Entmischung und Inferenz biomolekularer Netzwerke

Dissertation

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften

Der Fakultät Statistik

der Technischen Universität Dortmund vorgelegt von

Jakob Jan Wieczorek

aus Tarnowitz (Polen)

Abgabedatum: 4. Februar 2015

Tag der mündlichen Prüfung: 9. Juli 2015

Gutachter: Prof. Dr. Katja Ickstadt

Prof. Dr. Jörg Rahnenführer

Danksagung

An erster Stelle gilt mein Dank Frau Prof. Dr. Katja Ickstadt für die Möglichkeit in diesem spannenden Themenfeld der Systembiologie mitforschen und promovieren zu können und auch für die gute Betreuung und vielfache Hilfestellung.

Ebenso danken möchte ich allen Mitgliedern der Fakultät Statistik der Technischen Universität Dortmund, die zu jeder Zeit ein offenes Ohr für mich hatten und mir immer ein gutes Feedback gaben. Besonders danken möchte ich Dr. Marco Grzegorczyk, auf dessen Vorarbeit und reichen Erfahrungsschatz ich meine Forschung stützen konnte, Martin Schäfer, mit dem ich das Büro zu teilen das Glück hatte, Prof. Dr. Holger Schwender für die vielen guten Diskussionen, Jana Fruth für ihre fortwährende Hilfe und Unterstützung sowie Dr. Bernd Bischl für seine Beratung bei diversen Programmierproblemen. Nicht unerwähnt bleiben sollen auch meine beiden Lektoren Max Wornowizki und Dr. Anita Thieler.

Weiter möchte ich mich bei den Mitarbeitern des Max-Planck-Instituts für molekulare Physiologie in Dortmund bedanken. Unsere Kooperation hat mir bei dem Verständnis der biologischen Zusammenhänge, welches für die erfolgreiche Durchführung der Arbeit unerlässlich war, sehr geholfen. Für die Unterstützung im Projekt zur Aufklärung der Zelladhäsionsstrukturen gilt mein Dank insbesondere Dr. Eli Zamir, Dr. Hernàn Grecco sowie Dr. Rahuman Sheriff. Für die gute Zusammenarbeit im Projekt zur Analyse des mating pathways der Hefe danke ich Dr. Christina-Maria Hecker, Johann Jarzombek sowie Prof. Dr. Philippe Bastiaens.

Mein ganz besonderer Dank gilt dem Bundesministerium für Bildung und Forschung, ohne dessen Vertrauen in das Projekt und dessen finanzielle Unterstützung diese Arbeit nicht möglich gewesen wäre (Förderkennzeichen 0315507).

Inhaltsverzeichnis

Abbildungsverzeichnis

1. Einleitung

In den letzten Jahren wurde die klassische Biologie, profitierend vom Fortschritt der Mess- und Analysemethoden, um neue Zweige erweitert. Diese gewinnen stetig an Beachtung und bilden den Schwerpunkt großer Forschungsprojekte. Eine herausragende Stellung nimmt hier die Systembiologie ein.

Dieser Forschungsbereich strebt danach, die verschiedenen Mechanismen, wie sie in lebenden Zellen und Geweben vorkommen, quantitativ zu erfassen und deren Zusammenhänge mathematisch abzubilden sowie statistisch zu modellieren. Das Ziel dieser Bestrebungen ist es das Verständnis der untersuchten Strukturen zu vertiefen. Dabei sollen schrittweise erst einzelne Zellen dann Gewebe und schließlich ganze Organismen realitätsnah simuliert werden können. Auf diese Weise könnten, auf lange Sicht, in silico Experimente die Kosten und Risiken bei klinischen Studien deutlich reduzieren und vielleicht sogar klassische Experimente mit Beteiligung von Menschen oder Tieren ganz ersetzen. Bis dieses Ziel jedoch erreicht werden kann, müssen noch viele offene Fragen geklärt werden. So mangelt es beispielsweise an zuverlässigen Methoden, um aus einer Menge von potentiellen Reaktionspartnern in einer Zelle diejenigen zu identifizieren, welche gewöhnlich nicht miteinander interagieren, bei Vorliegen bestimmter Bedingungen oder äußerer Reize jedoch ihr Reaktionsverhalten ändern und in Interaktion treten. Derartige Änderungen können für die Zelle und das betroffene Gewebe mitunter drastische Konsequenzen haben und den Übergang von Normalzustand zu Krankheit erklären. Ein Verfahren, welches aus beobachteten Daten diese identifizieren könnte, wäre von großem Nutzen. In Rahmen dieser Arbeit werden Probleme aus diesem Themenfeld aufgegriffen und Lösungsvorschläge präsentiert. Insbesondere wird dabei auf die Modellierung von Interaktionen zwischen Objekten des biochemischen Stoffwechsels als Netzwerke eingegangen.

Das Konzept der Netzwerkinferenz, d.h. komplexe Systeme als Netzwerke zu beschreiben, um sie so zu vereinfachen und in der Folge besser untersuchen und begreifen zu können, ist weit verbreitet. In der Systembiologie kommt es z.B. bei der Erforschung des Metabolismus (Stower, 2014), der Homöostase (Donghyeon et al., 2013), der Morphogenese (Rabus et al., 2014) oder der Signalübertragung (Madhamshettiwar et al., 2012) zur Anwendung.

Zunehmend wird es aber auch abseits der Naturwissenschaften mit großen Erfolg eingesetzt. Man verwendet es beispielsweise in der Meinungsforschung, um die Faktoren, die zu einem bestimmten Urteil/Kaufentscheidung führen, zu sortieren, in der Betriebswirtschaftslehre, um Einflüsse auf den Markt zu beurteilen sowie in der Logistik zur Kontrolle von Waren- und Personenströmen (Armbruster et al., 2005; Newman, 2010; Jackson, 2010; Becker et al., 2013).

Der Begriff Netzwerkinferenz umfasst zwei Aspekte. Einerseits ist darunter die Analyse gegebener Netzwerkstrukturen gemeint (Brandes und Erlebach, 2005), wie zum Beispiel die Suche nach zentralen Knoten in sehr großen Systemen, oder die Bestimmung von Teilen eines Netzwerks mit bestimmten Eigenschaften. Anderseits beschreibt der Begriff Netzwerkinferenz auch die Rekonstruktion unbekannter Netzwerkstrukturen aus Beobachtungen. In dieser Arbeit wird der zweite Aspekt adressiert.

Als Ausgangspunkt dienen hierbei statistische Modelle in ihrer allgemeinen Form. Formal unterscheidet man zwischen parametrischen Modellen $\left(\mathfrak{M}, \mathfrak{A}, (P_\vartheta)_{\vartheta\in\Theta}\right)$ und nicht parametrischen Modellen $\left(\mathfrak{M}, \mathfrak{A}, \mathcal{P}\right)$, dabei steht $\left(\mathfrak{M}, \mathfrak{A}\right)$ für einen Messraum, $(P_\vartheta)_{\vartheta\in\Theta}$ für eine Familie von Wahrscheinlichkeitsverteilungen und $\mathcal{P}$ für eine Menge von Wahrscheinlichkeitsverteilungen. Vereinfacht ausgedrückt beschreiben statistische Modelle den Zusammenhang zwischen Variablen. Die Beschreibung kann sowohl die Art als auch die Stärke des Zusammenhanges wiedergeben. Netzwerke sind eine spezielle Form statistischer Modelle. Hier ist die Interaktionsstruktur der Zufallsvariablen ein wichtiges, oft sogar das Hauptziel der Analyse. Im Gegensatz zum allgemeinen Fall gibt es bei der Netzwerkinferenz im Vorhinein keine Abgrenzung zwischen beziehungsweise keine Festlegung von Einfluss- und Zielvariablen, wie das beispielsweise in klassischen Regressionsmodellen ($y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots$) mit einer abhängigen Variablen (y) und einer oder mehreren unabhängigen Variablen ($x_1, x_2, \ldots$) der Fall ist. In einem Netzwerk $\mathcal{N}$ kann eine Variable beide Rollen einnehmen. So hängt im Netzwerk, welches in Abbildung 1 links abgebildet ist, X_3 von X_1 ab, beeinflusst aber selbst wiederum X_4. Sowohl im parametrischen als auch im nichtparametrischen Fall hat $\mathcal{N}$ einen bedeutenden Einfluss auf die Wahrscheinlichkeitsverteilungen $\widetilde{P}$ aus $\left(\mathfrak{M}, \mathfrak{A}, (P_\vartheta)\right)_{\mathcal{N}\in\mathfrak{N}}$ bzw. $\widetilde{\mathcal{P}}$ aus $\left(\mathfrak{M}, \mathfrak{A}, \mathcal{P}\right)_{\mathcal{N}\in\mathfrak{N}}$. Dabei ist $\mathfrak{N}$ die Menge aller Netzwerke. Die Wahl der Modellklasse ist nicht unproblematisch. Wenn genügend

Vorwissen über das untersuchte System vorliegt, so dass ein zutreffendes Annahmengerüst aufgestellt werden kann, um ein parametrisches Modell zu untermauern, ist ein solches meist deutlich effizienter. Im Falle, dass nur wenig über das untersuchte System bekannt ist, haben sich nichtparametrische Ansätze bewährt, da hier das Spektrum der überprüften Modelle nicht beschränkt wird. So können beispielsweise die Parameter, welche das Netzwerkmodell spezifizieren, frei variieren und müssen nicht wie in parametrischen Modellen aus einer fest vorgegebenen Verteilungsfamilie stammen.

Innerhalb der Methoden zur Rekonstruktion unbekannter Netzwerkstrukturen wird zwischen statischen Methoden zur Analyse von festen Zeitpunkten und dynamischen Methoden zur Analyse von Zeitreihen, bestehend aus aufeinander folgenden Zeitpunkten $(t_1, \ldots, t_{\mathbf{T}})$, unterschieden (vergleiche Abbildung 1, rechts). Die dynamischen Methoden können wiederum zu der parametrischen $\left(\mathfrak{M}, \mathfrak{A}, (P_\vartheta)_t\right)$ oder der nichtparametrischen $\left(\mathfrak{M}, \mathfrak{A}, (\mathcal{P})_t\right)$ Klasse gehören. Es gibt zwar partielle Überschneidungen in den angewandten Konzepten zwischen den statischen und den dynamischen Methoden, dennoch sind diese nicht austauschbar und müssen problemspezifisch gewählt werden.

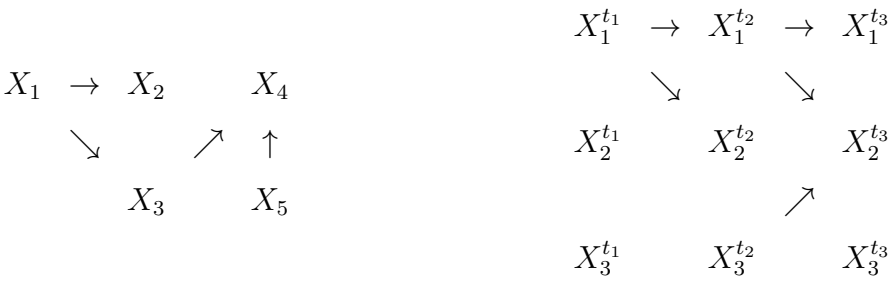


Abbildung 1: *Schematische Darstellung eines statischen Netzwerks mit fünf Variablen* $(X_1, \ldots, X_5)$ *(links) und eines dynamischen Netzwerks mit drei Variablen und drei Zeitpunkten* $(t_1, \ldots, t_3)$ *(rechts).*

Die Algorithmen zur Bestimmung von $\mathcal{N}$ sind sehr vielfältig, haben aber gemeinsam, dass die Schätzung mittels Optimierung einer geeigneten Funktion $\mathfrak{L}(\mathcal{N}|\vartheta, \text{Daten})$, welche für zutreffende $\mathcal{N}$ extreme Werte annimmt, erfolgt. Einige prominente Vertreter aus dem Bereich der dynamischen Netzwerke sind dynamische Bayessche Netzwerke (Friedman et al., 1998; Grzegorczyk et al., 2008; Mazur et al., 2009), dynamische Petri-Netze (Hardy und

Robillard, 2008; Sackmann et al., 2006) sowie gewöhnliche, partiell lineare und stochastische Differentialgleichungen (Chen et al., 1999; Mazur et al., 2009). Einen breiten Überblick sowie eine Besprechung der dynamischen Netzwerkinferenzmethoden bieten folgende Artikel der Fachliteratur: de Jong H. (2002); Markowetz und Spang (2007) sowie Zhao und Qiao (2013). Die für diese Arbeit formulierten Fragestellungen und Zielsetzungen bedingen eine Fokussierung auf Methoden, die im systembiologischen Kontext für die Inferenz von Netzwerken aus statischen Daten konzipiert wurden. Aus diesen Grund werden sie im Kapitel 2 eingehend besprochen.

Diese Arbeit thematisiert neben der Bestimmung der Interaktionsstruktur innerhalb eines Netzwerkes als eine der ersten auch den Aspekt der „vermischten Daten“. Diese Umschreibung meint das Problem, dass die betrachteten biochemischen Vorgänge im organischen Gewebe, zwischen und innerhalb der Zellen, sich mit hoher zeitlicher und räumlicher Überlappung abspielen. Damit kann nicht ausgeschlossen werden, dass auch bei Verwendung der empfindlichsten gegenwärtig verfügbaren Messtechnik Werte von z.B. kranken und gesunden Zellen in einem Datensatz unerkannt nebeneinander vorliegen. Modelle, die aus einer solchen inhomogenen Grundlage abgeleitet werden, sind mit einer hohen Wahrscheinlichkeit verzerrt und für keine der beiden Gruppen passend. Sämtliche darauf aufbauende Analysen unterliegen starker Unsicherheit. Dieses Problem ist in der Statistik, in seiner allgemeinen Form, als Problem der latenten Variablen zwar bekannt, im Zusammenhang mit Netzwerken aber kaum thematisiert worden. Bisherige Lösungsansätze beruhen überwiegend auf verschiedenen Formen der Datenvorverarbeitung mit dem Schwerpunkt auf Verfahren aus dem Bereich der Clusteranalyse. Dabei werden die Rohdaten vor der eigentlichen Netzwerkschätzung in Cluster aufgeteilt. Eine etwas elegantere Methode dem Vermischungsproblem zu begegnen ist das parallele Verwenden mehrerer Modelle. So könnte ein Modell die gesunden, ein zweites Modell die kranken Zellen behandeln und beide in Kombination die Gesamtpopulation beschreiben. Diese Art von statistischen Modellen, welche Variablen aus mehreren Grundgesamtheiten behandeln, nennt man Mischungsmodelle. Hierbei muss die Wahrscheinlichkeit, dass die Beobachtung aus einer bestimmten Grundgesamtheit stammt, oder, um bei dem Beispiel zu bleiben, die Wahrscheinlichkeit, dass eine kranke bzw. eine gesunde Zelle vorliegt, ebenfalls modelliert werden. Alle die-

se Verfahren erfordern jedoch, dass die Clusteranzahl bzw. die Anzahl der Modelle vor der Berechnung festgelegt wird. Das stellt ein Problem dar, da diese unbekannt ist und der Anwender somit gezwungen ist, Vorwissen einzubringen, welches nur in den seltensten Fällen zur Verfügung steht. Wird mit einer nicht zutreffenden Anzahl von im Datensatz vorkommenden Gruppen gearbeitet, ist eine korrekte Zuordnung nicht mehr möglich und die Analyseergebnisse sind unweigerlich verfälscht. Die in den sechziger Jahren bekannt gewordene Hierarchische Clusteranalyse nach Ward (1963) kann diesen Nachteil zwar etwas abschwächen, hier können verschiedene Clusterzahlen ohne eine neue Berechnung ausprobiert werden, das grundlegende Problem, die manuelle Festlegung der finalen Clusteranzahl für die weitere Analyse, ist jedoch noch vorhanden.

In dieser Arbeit wird eine Alternative zur bisherigen Modellierung vorgeschlagen. Dieser neue, Entmischung genannte, Ansatz gruppiert die Datenpunkte, zum Beispiel Zellen, anhand ihrer Interaktionsstruktur und bestimmt automatisch die Clusteranzahl. Dabei sind die Schätzung der Clusteranzahl und die der Netzwerkstruktur ($\mathcal{N}$) miteinander verflochten, wie die folgende, stark vereinfachte Funktion zeigt:

$$\mathcal{L}(\mathcal{N}|\vartheta, \text{Daten}) = \sum_{i \in \mathbb{N}} \mathcal{L}(\mathcal{N}_i|\vartheta_i, \text{Daten}_i) \ ,$$

mit $\text{Daten} = \bigcup_{i \in \mathbb{N}} \text{Daten}_i$ sowie $\text{Daten}_{i^*} \bigcap \text{Daten}_{i^{**}} = \emptyset$, für $i^* \neq i^{**}$. In dieser Arbeit erfolgt die Partitionierung des Datensatzes mittels Methoden aus der Bayesschen Nichtparametrik (Ghosh und Ramamoorthi, 2003). Insbesondere kommen der Dirichlet-Prozess sowie der Pitman-Yor-Prozess zum Einsatz. Die zur Berücksichtigung der verschiedenen Teildatensätze notwendige Flexibilität des Modells wird durch das Zurückgreifen auf das Konzept der Mischmodelle gewährleistet (Antoniak, 1974).

Nach einer allgemeinen Einführung in das Gebiet der Netzwerkinferenz mit einer Übersicht der wichtigsten statischen Methoden im zweiten Kapitel werden im dritten Kapitel die für diese Arbeit zentralen statistischen Verfahren beschrieben. Dabei wird sowohl auf bereits bekannte als auch auf im Rahmen dieser Arbeit neu entwickelte Verfahren eingegangen. Unter anderem wird ein neuer Aspekt des Umgangs mit Netzwerken aufgegriffen und ein Verfahren für die Clusterung von Netzwerken vorgestellt. Im vierten Kapitel wird die Leistungsfähigkeit der zuvor erarbeiteten Methoden im Rahmen einer Simulations-

studie untersucht und mit Referenzmethoden verglichen. Das sechste Kapitel behandelt die Untersuchung des als mating pathway bekannten Stoffwechselweges in der Hefe, bei dem die neu entwickelten Verfahren mit Erfolg zur Anwendung kommen. Die Messungen des Stoffwechselweges liegen, aufgrund einer technischen Beschränkung, jedoch nicht in der benötigten Auflösung vor. Deshalb müssen diese zuvor analysiert werden. Zu diesen Zweck wird im fünften Kapitel ein Schätzer entwickelt, der eine deutliche Verbesserung der gängigen Analysepraxis dieser Art von Daten darstellt. Abschließend folgt das siebte Kapitel mit Diskussion und Ausblick.

2. Überblick über die statistischen Methoden der Netzwerkinferenz

Im ersten Teil des folgenden Kapitels werden die für das Verständnis der Arbeit essentiellen Begriffe kurz erklärt. Die eingeführten Abkürzungen und Notationen können auf Seite 136 im Stichwortverzeichnis nachgeschlagen werden. Dem folgt ein Überblick der in der Systembiologie gängigen Methoden für die Inferenz statischer Netzwerke.

2.1. Grundlegende Definitionen

Der Begriff **Netzwerk** ist im Sprachgebrauch mehrfach belegt. In dieser Arbeit wird darunter das statistische Modell einer Interaktionsstruktur für eine fest definierte Menge von Objekten verstanden. Dieses umfasst grundsätzlich die Darstellung des Netzwerks als Graph. Ist im vorliegendem Text von einem biomolekularen oder biochemischen Netzwerk die Rede, so sind die in der Zelle ablaufenden, teils unbekannten Interaktionen und biochemischen Prozesse gemeint.

Ein **Graph** ist definiert als Tupel $\mathcal{G} = (V, E)$ aus einer endlichen Menge von Knoten V $(V_1, \ldots, V_d)$, welchen eineindeutig Zufallsvariablen X $(X_1, \ldots, X_d)$ zugeordnet werden können, und aus einer Menge von Kanten $E \subset V \times V$. Es wird zwischen gerichteten und ungerichteten Kanten unterschieden. Bei einer gerichteten Kante, bezeichnet mit $\vec{X}_{j^*j}$, wird der Knoten X_{j^*} vom Knoten X_j beeinflusst. In diesem Fall wird X_j als Nachfahre von X_{j^*} und aller weiterer Knoten, die auf X_j weisen, bezeichnet. Die Menge dieser Knoten wird als **Elternmenge** $\mathrm{pa}_{\mathcal{G}}(X_j)$, oder abgekürzt $\mathrm{pa}(X_j)$, bezeichnet. Hingegen steht $\overset{\leftrightarrow}{X}_{j^*j}$ für eine ungerichtete Kante. Diese symbolisiert lediglich, dass die Knoten X_{j^*} und X_j von einander abhängen. Für eine umfassende Einführung in dieses Gebiet kann das Buch von Diestel (2010) herangezogen werden.

Man nennt (V, E) einen **gerichteten azyklischen Graphen**, häufig auch nach der englischen Benennung directed acyclic graph als DAG bezeichnet, falls alle Kanten in E gerichtet sind und er keine Kreise enthält, also für jeden Knoten $X_* \in V$ gilt, dass kein Pfad,

d.h. eine Folge von gerichteten Kanten, existiert, welche von X_* ausgeht und dort wieder hinführt. Die Abbildung 2 auf Seite 9 zeigt beispielhaft einen DAG eines (Bayesschen) Netzwerks.

Bayessche Netzwerke (BN, Pearl (1985)) sind eine besondere Klasse von statistischen Modellen. Sie bestehen aus einem DAG ($\mathcal{G}$), in dem die Knoten Zufallsvariablen und die Kanten bedingte Abhängigkeiten zwischen den Variablen repräsentieren. Die Wahrscheinlichkeitsverteilung jedes Knotens X_j wird dabei durch eine bedingte Wahrscheinlichkeitsverteilung in Abhängigkeit von der Elternmenge $\mathrm{pa}_{\mathcal{G}}(X_j)$ beschrieben. Zudem lässt sich die gemeinsame Verteilung der Knoten analog zu der in dem DAG kodierten Abhängigkeitsstruktur faktorisieren, so dass $p(X_1, \ldots, X_d) = \prod_{j=1}^{d} p(X_j|\mathrm{pa}_{\mathcal{G}}(X_j))$ gilt, vergleiche Abbildung 2. Diese für die vorliegende Arbeit zentralen Modelle werde in Abschnitt 2.6 eingehend behandelt.

Für das Bearbeiten von DAGs werden sogenannte **Kantenoperationen**, im englischen unter single edge operations geläufig, verwendet. Es wird zwischen dem Entfernen, dem Hinzufügen oder dem Umkehren einer Kante unterschieden. Im Kontext dieser Arbeit sind diese Operationen nur dann zulässig, wenn der resultierende Graph wieder ein DAG ist. Alle DAGs, die durch eine einzige Kantenoperation aus einem DAG* potentiell entstehen können, werden als **Nachbargraphen** von DAG* bezeichnet, ihre Menge wird mit M_{DAG^*} notiert.

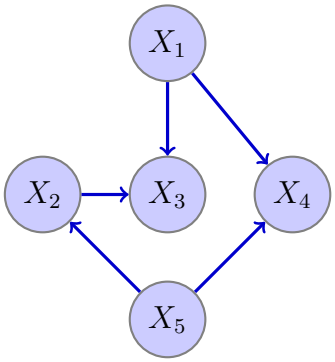
Als **Datenmatrix** wird die Matrix $\mathcal{X} \in \mathbb{R}^{n \times d}$ bezeichnet, bestehend aus $n \in \mathbb{N}$ Beobachtungen eines biochemischen Netzwerkes mit $d \in \mathbb{N}$ Knoten, deren Interaktionsstruktur modelliert werden soll. Die Zeilen von $\mathcal{X}$ werden mit $\boldsymbol{x}_1', \ldots, \boldsymbol{x}_n'$ notiert, wobei $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})'$, dabei steht v' für die Transponierte eines Vektors v. Damit steht x_{ij} für die i-te Beobachtung der j-ten Variablen (des j-ten Knotens). Im systembiologischen Kontext werden Objekte, die miteinander interagieren, häufig als **Spezies** bezeichnet.

Ein **Allokationsvektor** $\boldsymbol{l} \in \mathbb{N}^{n \times 1}$ gibt für jede Beobachtung eine Gruppenzugehörigkeit an. Dabei kann es sich um die wahre Gruppenzugehörigkeit handeln, wenn diese z.B. durch das Experimentdesign bekannt ist. Er kann aber auch das Ergebnis einer Schätzung wiedergeben. Diese Gruppen $(1, \ldots, h, \ldots, N)$ haben je nach Kontext unterschiedliche Interpreta-

tionen. Im Zuge der später in Abschnitt 3.2 beschriebenen nichtparametrischen Bayesschen Netzwerke stehen die Gruppen für N Netzwerke, aus denen die Beobachtungen stammen. Gleichzeitig ist der Allokationsvektor aber auch eine Partitionierung der Beobachtungen, so dass eine Betrachtung der Gruppen als Cluster sinnvoll ist. Diese werden ebenfalls mit h indiziert. Auf die jeweilige Verwendung wird im Text hingewiesen.

Eine **Adjazenzmatrix** $\mathcal{A} \in \{0,1\}^{d \times d}$, auch als Nachbarschaftsmatrix bekannt, ist eine mathematische Repräsentation von Graphen bzw. DAGs.
Der Eintrag $\mathcal{A}_{j^*j} = 1$ bedeutet, dass Knoten X_{j^*} zu der Elternmenge vom Knoten X_j gehört, während $\mathcal{A}_{j^*j} = 0$ bedeutet, dass keine Kante vom Knoten X_{j^*} zu Knoten X_j führt. (Der umgekehrte Fall ist aber nicht ausgeschlossen.) In Abbildung 2 auf Seite 9 ist eine Beispieladjazenzmatrix für ein einfaches Netzwerk dargestellt.

$$
\begin{aligned}
&p(X_1, X_2, X_3, X_4, X_5) \\
&= p(X_1) \cdot p(X_2|X_5) \\
&\quad \cdot p(X_3|X_1, X_2) \\
&\quad \cdot p(X_4|X_1, X_5) \cdot p(X_5)
\end{aligned}
$$



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

Abbildung 2: *Graph eines Bayesschen Netzwerkes mit fünf Knoten $X_1, \ldots, X_5$ mit zugehöriger Faktorisierung der gemeinsamen Verteilung (links) sowie der korrespondierenden Adjazenzmatrix $\mathcal{A}$ (rechts). Die Elternmenge des Knotens X_3 sind die Knoten X_1 und X_2, $pa(X_3) = (X_1, X_2)$, die Knoten X_1 und X_5 haben keine Eltern.*

In den folgenden Abschnitten wird ein Überblick über die wichtigsten Klassen von Netzwerkinferenzmethoden für statische Datenstrukturen gegeben.

2.2. Methoden zur Schätzung der Graphenstruktur, die auf der Kovarianzmatrix beruhen

Die Interaktionsstruktur in einem Netzwerk anhand der Kovarianzmatrix aufzudecken ist eine naheliegende Herangehensweise. Dementsprechend gehören diese Verfahren mit zu den ersten, die bereits Angang der 90er Jahre nicht nur bekannt waren, sondern auch praktische Anwendung fanden (Simonson und Perahia, 1992). Traditionell werden diese im Zusammenhang mit Microarray-Messungen verwendet, in denen die Zahl der betrachteten Gene deutlich höher als die Zahl der Messwiederholungen ist. Es existieren zahlreiche, auf die konkreten Experimente angepasste Modifikationen und Abwandlungen der zentralen Mechanismen. Einen Überblick bieten die Publikationen von Krämer et al. (2009), Tenenhaus et al. (2010) sowie Stifanelli et al. (2013). Das zugrunde liegende Prinzip soll anhand der erstgenannten vorgestellt werden.

Sei $\mathcal{X} \in \mathbb{R}^{n \times d}$ die Datenmatrix mit spaltenweise zentrierten Einträgen, so dass der Mittelwert von jedem Knoten bei null und die Standardabweichung bei eins liegt. Die Spalten beschreiben die Gene, die Zeilen die Beobachtungen (die Microarrays). Zur unverzerrten Schätzung der $d \times d$ Kovarianzmatrix Σ kann auf

$$\hat{\Sigma} = \frac{1}{n-1} \mathcal{X}^T \mathcal{X}$$

zurückgegriffen werden. In Fällen, in denen $\hat{\Sigma}$ invertierbar ist, kann eine unverzerrte Schätzung der partiellen Korrelationen zwischen Gen j^* und j aus

$$\hat{\rho}_{j^*j} = -\frac{\hat{\omega}_{j^*j}}{\sqrt{\hat{\omega}_{j^*j^*}\hat{\omega}_{jj}}}$$

mit $\hat{\Sigma}^{-1} = (\hat{\omega}_{j^*j})$ erhalten werden. Dieser Fall tritt in der Praxis aber nur selten ein und die inverse Kovarianzmatrix muss geschätzt werden. Schäfer und Strimmer (2005) schlagen dafür einen regularisierten Schätzer

$$\hat{\Sigma}_\lambda = \lambda \hat{T} + (1-\lambda)\hat{\Sigma}$$

vor. Dieser besteht aus der zuvor beschriebenen Schätzung von $\hat{\Sigma}$ und aus der Matrix $\hat{T}$, einer vereinfachten, restringierten Version von $\hat{\Sigma}$. Der Parameter $\lambda \in [0, 1]$ regelt den

Grad der Regularisierung. Beide Schätzer haben Vorzüge und Nachteile: $\hat{\Sigma}$ ist zwar erwartungstreu, hat aber tendenziell eine hohe Varianz. $\hat{T}$ zeichnet sich durch eine geringe Varianz aus, ist dafür aber verzerrt. Durch die konvexe Kombination entsteht ein regularisierter Schätzer, der genauere Ergebnisse (also einen geringeren MSE) erzielt als seine einzeln angewendeten Komponenten. In das Netzwerkmodell werden diejenigen Kanten aufgenommen, die zu Knotenpaaren korrespondieren deren Kovarianz einen Schwellenwert überschreitet. In der Literatur wird 0.2 als Schwellenwert vorgeschlagen.

2.3. Methoden zur Schätzung der Graphenstruktur, die auf partieller Korrelation oder auf Regressionsverfahren beruhen

Auch in der Methodenklasse zur Schätzung der Graphenstruktur, welche partielle Korrelation oder Regressionsverfahren nutzen, liegt eine große Vielfalt vor. Einen groben Überblick bieten die Publikationen von Krämer et al. (2009); Meinshausen und Bühlmann (2006); Kalisch und Bühlmann (2007); Banerjee et al. (2008); Wainwright et al. (2007); Kim et al. (2013) und Ravikumar et al. (2010). Anhand der Publikation von Krämer et al. (2009) können die hier angewandten Methoden vorgestellt werden.

Sei $\mathcal{X}$, wie im vorigen Abschnitt, die spaltenweise zentrierte Datenmatrix. Ausgangspunkt für die Schätzung der Graphenstruktur ist die klassische Kleinste-Quadrate-Schätzung. Für jeden der d betrachteten Knoten wird ein lineares Regressionsmodell aufgestellt. Das Verhalten des j^*-ten Knotens wird aus den übrigen $d-1$ Knoten erklärt,

$$X_{j^*} = \sum_{j \neq j^*} \beta_j^{(j^*)} X_j + \epsilon \ , \ \text{für } j^* = 1, \ldots, d \ ,$$

wobei ϵ für das unabhängig identisch normalverteilte Rauschen steht. Da die Daten zentriert sind, muss der Achsenabschnitt im Modell nicht berücksichtigt werden. Die Kleinste-Quadrate-Schätzung $\hat{\boldsymbol{\beta}}^{(j^*)} = (\beta_1^{(j^*)}, \ldots, \beta_{j^*-1}^{(j*)}, \beta_{j^*+1}^{(j^*)}, \ldots, \beta_d^{(j^*)})'$, $j^* = 1, \ldots, d$ des Vektors der Regressionskoeffizienten ergibt sich aus der Lösung des Optimierungsproblems

$$\hat{\boldsymbol{\beta}}^{(j^*)} = \underset{\boldsymbol{\beta}^{(j^*)} \in \mathbb{R}^{d-1}}{\operatorname{argmin}} \left\| \mathcal{X}^{(j^*)} - \mathcal{X}^{(\backslash j*)} \boldsymbol{\beta}^{(j^*)} \right\|_2 = \left(\mathcal{X}^{(\backslash j*)t} \mathcal{X}^{(\backslash j*)} \right)^{-1} \mathcal{X}^{(\backslash j^*)t} \mathcal{X}^{(\backslash j^*)}. \qquad (1)$$

Dabei ist $\mathcal{X}^{(j^*)} \in \mathbb{R}^n$ die j^*-te Spalte von $\mathcal{X}$ und $\mathcal{X}^{(\setminus j^*)} \in \mathbb{R}^{n \times (d-1)}$ steht für die Matrix $\mathcal{X}$ ohne die j^*-te Spalte. Ferner ist $\boldsymbol{\beta}^{(j^*)} = (\beta_1^{(j^*)}, \ldots, \beta_{j^*-1}^{(j*)}, \beta_{j^*+1}^{(j^*)}, \ldots, \beta_d^{(j^*)})'$. Die partielle Korrelation zwischen den Genen j^* und j wird geschätzt aus

$$\hat{\rho}_{j^*j} = \text{sign}\left(\hat{\beta}_j^{(j^*)}\right) \sqrt{\hat{\beta}_j^{(j^*)} \hat{\beta}_{j^*}^{(j)}}. \tag{2}$$

Dabei steht sign() für die Vorzeichenfunktion. Da $\hat{\beta}_j^{(j^*)}$ und $\hat{\beta}_{j^*}^{(j)}$ das gleiche Vorzeichen haben (vergleiche Krämer et al. (2009)), ist dieser Ausdruck wohldefiniert.
Aufgrund der häufig auftretenden Situation $d \gg n$ wird der klassische Ansatz für Netzwerkschätzung zur sogenannten regularisierten Regression ausgebaut. Der Kleinste-Quadrate-Schätzer aus Gleichung 1 wird um einen additiven Strafterm $\mathcal{S}(\boldsymbol{\beta})$ erweitert, so dass der Schätzer aus Gleichung 2 mit zuverlässigeren, im Sinne von weniger verzerrten, Ausgangswerten arbeiten kann.
Hoerl und Kennard (2000) schlagen vor, die Ridge-Regression mit dem Strafterm

$$\mathcal{S}(\boldsymbol{\beta}) = \lambda \|\boldsymbol{\beta}\|_2^2 = \lambda \sum_{j^*} \beta_{j^*}^2$$

zu verwenden. Hingegen schlagen Meinshausen und Bühlmann (2006) den Lasso-Ansatz vor (Tibshirani, 2011). Hier hat der Strafterm die Form

$$\mathcal{S}(\boldsymbol{\beta}) = \lambda \|\boldsymbol{\beta}\|_1 = \lambda \sum_{j^*} |\beta_{j^*}| \ .$$

Der Einfluss des Strafterms kann über den Parameter $\lambda > 0$ beeinflusst werden. Zur Findung der passenden Einstellung von λ können Kreuzvalidierungsverfahren herangezogen werden. Anregungen den Regressionsansatz auszubauen, bieten die Arbeiten von Zou (2006) oder Zhou et al. (2009) an. Diese regen an für große Netzwerke mit mehreren hundert Knoten mehrstufig vorzugehen. Für die Aufstellung des finalen Modells sollen Einflüsse von Knoten (β_j), welche in den Modellen der Zwischenstufen keinen oder nur geringen Beitrag zum modellierten Knoten (X_{j^*}) gezeigt haben, nicht berücksichtigt werden. Methodisch wird die Verwendung von adaptiven Lasso-Verfahren kombiniert mit Schwellenwert-Verfahren vorgeschlagen.

2.4. Methoden zur Schätzung der Graphenstruktur, die auf Konzepten aus der Informationstheorie beruhen

Die Netzwerkinferenzmethoden, welche auf der Informationstheorie basieren, stellen eine weitere große Methodenklasse dar (Shoudan, 1998; Butte und Kohane, 2000; Wentao et al., 2006). Eine der bedeutendsten Gruppen innerhalb dieser sind jene Verfahren, welche die sogenannte mutual information verwenden. Die elementaren Prinzipien, die hier zur Anwendung kommen, werden anhand der Arbeiten von Basso et al. (2005) sowie Margolin und Califano (2007) vorgestellt.

Das Verfahren verwendet die unterschiedlichen Entropien der Spalten von $\mathcal{X}$.
Ist generell X eine diskrete Variable, so ist die Entropie von X wie folgt definiert:

$$S(X) = -\sum_i p(X_i) \log p(X_i) \ ,$$

wobei $p(X_i)$ die diskrete Wahrscheinlichkeit der Ausprägung X_i der Variable X darstellt. Analog wird für stetige Variablen

$$S(X) = -\int_{\text{Träger}(X)} f(X) \log f(X) dX$$

verwendet, wobei $f(X)$ für die Dichtefunktion steht. Die gemeinsame Entropie zweier Variablen X und X^* ist im diskreten Fall mit der gemeinsamen Verteilung $p(X_i, X^*_{i^*})$ gegeben durch

$$S(X, X^*) = -\sum_i \sum_{i^*} p(X_i, X^*_{i^*}) \log p(X_i, X^*_{i^*}) \ .$$

Im stetigen Fall mit gemeinsamer Dichte $f(X, X^*)$ hat die gemeinsame Entropie von X und X^* folgende Form

$$S(X, X^*) = -\int_{\text{Träger}(X)} \int_{\text{Träger}(X^*)} f(X, X^*) \log f(X, X^*) dX dX^* \ .$$

Die mutual information zwischen zwei Variablen X und X^* ist dann definiert als

$$I_{X,X^*} = S(X) + S(X^*) - S(X, X^*) \ .$$

Für die Schätzung der Entropie schlagen Basso et al. (2005) folgenden Ansatz vor. Unter Ausnutzung der Eigenschaft, dass die mutual information invariant gegenüber umkehrbaren Transformationen ist, kommt die sogenannte copula Transformation (Joe, 1997) zum

Einsatz. Diese projiziert die $\Im$ Messungen jedes Knotens auf abstandsgleiche reelle Zahlen im Intervall $[0, 1]$. Die transformierte Variable X wird mit $X^{'}$ notiert. Die ursprüngliche Reihenfolge wird dabei nicht verändert. Als Folge dieser Transformation werden $S(X^{'})$ und $S(X^{*'})$ zu Konstanten mit dem Wert null. Im Ausdruck I_{X,X^*} fallen diese Terme somit weg und es muss nur noch $S(X^{'}, X^{*'})$ bestimmt werden. Basso et al. (2005) empfehlen dafür die Verwendung eines Gaußschen Kerndichteschätzers mit

$$I_{X',X^{*'}} = \frac{1}{\Im} \sum_i \log \left[\frac{p(X_i, X_i^*)}{p(X_i)p(X_i^*)} \right] ,$$

wobei i die beobachteten Ausprägungen indiziert.
Die Größen $p(X_i),\ p(X_i^*)$ und $p(X_i, X_i^*)$ sind wie folgt definiert:

$$p(X_i) = \frac{1}{\sqrt{2\pi}\Im d_1} \sum_{\mathrm{i}} e^{\frac{[X_i - X_{\mathrm{i}}]^2}{2d_1^2}} \quad , \quad p(X_i, X_i^*) = \frac{1}{2\pi\Im d_2^2} \sum_{\mathrm{i}} e^{\frac{[X_i - X_{\mathrm{i}}]^2 + [X_i^* - X_{\mathrm{i}}^*]^2}{2d_1^2}} \quad .$$

Die optimalen Einstellungen der Glättungsparameter d_1 und d_2 können mittels Monte-Carlo-Simulation bestimmt werden. Überschreitet I_{X,X^*} für zwei Zufallsvariablen (X, X^*) einen bestimmten Schwellenwert, so gelten die korrespondierenden Knoten im Netzwerkmodell als verbunden. Die Wahl des Schwellenwertes hängt von der konkreten Situation ab. Ähnlich dem Korrelationskoeffizienten misst die mutual information den Grad der statistischen Abhängigkeit zwischen zwei Variablen. Sie hat jedoch den Vorteil, invariant gegenüber Umparametrisierungen zu sein. Ferner ist sie genau dann ungleich null, wenn die Variablen statistisch abhängig sind (Shannon, 1948). Eine der ersten Anwendungen der mutual information in Zusammenhang mit Netzwerkinferenz ist die Arbeit von Butte und Kohane (2000), während die Publikation von Margolin et al. (2006) die aktuellen Entwicklungen zeigt.

2.5. Methoden zur Schätzung der Graphenstruktur aus der Gruppe der Petri-Netze

Petri-Netze (PN), benannt nach den Informatiker Carl Adam Petri (Peterson, 1978), sind ein weit verbreitetes Werkzeug um komplexe, parallel ablaufende Vorgänge in größeren Systemen zu beschreiben und zu analysieren. Das ursprüngliche Verfahren wurde in zahlreichen Varianten und Abwandlungen an neue Probleme angepasst. Beispiele für aktuelle

Anwendungen sind die Arbeiten von Ruths et al. (2008), Werhli (2012) und Berestovsky et al. (2013). Einen guten allgemeinen Überblick bieten die Publikationen von Chaouiya (2007) und Silva (2013). Eine ausführliche Besprechung der PN im biologischen Kontext enthält das Buch von Koch et al. (2011). Aktuell kommen PN vermehrt zum Einsatz, wenn bei bekannten Netzwerken die Art und die Stärke der Interaktion zwischen den Spezies erforscht werden soll. PN liefern aber auch bei der Rekonstruktion der Netzwerkstruktur gute Ergebnisse, wie die Publikation von Küffner et al. (2010) zeigt, an der sich der folgende Abschnitt orientiert.

Ein PN ist ein gewichteter, gerichteter, bipartiter Graph. Die Eigenschaft bipartit bedeutet, dass der Graph aus zwei disjunkten Knotenmengen besteht, wobei die eine die sogenannten Stellen (places) und die andere die Transitionen (transitions) umfasst. Die zwei Mengen repräsentieren verschiedene Arten von Knoten. In der Systembiologie stehen die Stellen für Gene oder Proteine und die Transitionen für biochemische Reaktionen, welche zwischen ihnen stattfinden.

Ein PN ist weiter dadurch charakterisiert, dass gerichtete Kanten Stellen und Transitionen verbinden. Eine Kante ist entweder eine Inputkante (Verbindung zwischen einer Stelle und einer Transition) oder eine Outputkante (Verbindung zwischen einer Transition und einer Stelle). Es gibt keine direkten Verbindungen von Stelle zu Stelle oder von Transition zu Transition. Die gerichteten gewichteten Kanten bestimmen nicht nur die Interaktionspartner, sondern auch die Charakteristika der Reaktion, wie z.B. die Geschwindigkeit oder den Verbrauch der reagierenden Spezies.

Der Rekonstruktionsvorgang des Netzwerks erfolgt iterativ gemäß dem folgendem Schema: Initialisiere anhand einer Menge von PN (die Population), welche entweder zufällig oder basierend auf Expertenwissen erstellt wird.

1. Wähle für jedes PN in der Population zufällig sogenannte Operationen aus. Diese ändern entweder die Topologie (durch Entfernen oder Hinzufügen von Kanten) oder die Art und Stärke der Interaktion (durch Variation der Parameter). Es ist ebenfalls möglich, ähnlich wie bei evolutionären Algorithmen, Teile verschiedener PN innerhalb der Population zu kombinieren. Dies beschleunigt die Schätzung deutlich.

2. Verwende die entstandenen PN, um unter Verwendung der Beobachtungen Referenzdaten $Y \in \mathbb{R}^{n \times d}$ zu generieren.

3. Bestimme mit Hilfe der Referenzdaten die Qualität der in Schritt 1 entstandenen Modelle mittels der Bewertungsfunktion dist(PN) (vergleiche Gleichung 3).

4. Dünne die Population der PN aus, so dass nur noch Modelle, die eine bestimmte Mindestqualität erfüllen, in die nächste Iteration der Schätzung eingehen.

Wiederhole Schritt 1-4 bis zu einer festgelegten Iterationszahl oder bis keine Verbesserung mehr erzielt werden kann.

In der Arbeit von Küffner et al. (2010) wird Schritt 4 durch die Anwendung von simulated annealing ergänzt. So kann sichergestellt werden, dass der Raum der möglichen PN umfassend abgesucht wird und dass das Verfahren lokale Maxima überwinden kann.

Die Bewertungsfunktion dist() basiert auf dem Bravais-Pearson Korrelationskoeffizienten ρ zwischen dem Vektor der Beobachtungen des j-ten Knotens, enthalten in der j-ten Spalte von $\mathcal{X}$ ($\mathcal{X}^{(j)}$) und dem Vektor der für diesen Knoten generierten Referenzwerte, enthalten in der j-ten Spalte von Y ($Y^{(j)}$). Die Bewertungen der einzelnen Knoten fließen gemittelt in die Bewertungsfunktion für das gesamte PN ein,

$$\text{dist}(PN) = \left[1 - \frac{1}{d} \sum_{j=1}^{d} \rho(\mathcal{X}^{(j)}, Y^{(j)}) \cdot \zeta_j \right]^2 + \text{reg} \cdot |PN| \ . \tag{3}$$

Über den Parameter reg $\in [0, 1]$ kann die Komplexität des Modells (Anzahl der Knoten) im Strafterm berücksichtigt werden. Dies trägt dazu bei, Überanpassung zu vermeiden. Der Vektor $\zeta \in [0, 1]^d$ kann zusätzlich Vorwissen in die Funktion einbringen, zum Beispiel ob bestimmte Gene hoch- bzw. herabreguliert wurden. Kleine Werte von dist zeigen eine gute Übereinstimmung zwischen den Referenzen und den Beobachtungen an. Dies lässt auf ein zutreffendes Netzwerkmodell schließen.

2.6. Methoden zur Schätzung der Graphenstruktur basierend auf Bayesschen Netzwerken

Eine bedeutende Gruppe von Netzwerkinferenzkonzepten fußt auf dem von Pearl (1985) geprägten Begriff der Bayesschen Netzwerke. Der ursprüngliche Ansatz wurde von vielen Wissenschaftlern, die auf dem Gebiet der Systembiologie forschen, aufgegriffen und in zahlreichen Varianten eingesetzt. Folgende Publikationen können für einen Überblick herangezogen werden: Cooper und Herskovits (1992); Friedman et al. (2000); Needham et al. (2006); Zhao und Qiao (2013).

Von einen Bayesschem Netzwerk $(p, \mathcal{G})$ spricht man, wenn p ein System von bedingten Wahrscheinlichkeitsverteilungen $p\big(X_j|\mathrm{pa}_{\mathcal{G}}(X_j)\big)$ ist, welche mit den Knoten eines DAGs $\mathcal{G}$ assoziiert sind (Pearl, 1985). Ein Beispiel für ein einfaches Bayessches Netzwerk ist in Abbildung 2 auf Seite 9 dargestellt. Sind zusätzlich alle Verteilungen aus p normalverteilt und der Gestalt

$$X_j|\mathrm{pa}_{\mathcal{G}}(X_j) \sim N\Big(\mu_j + \sum_{\mathbf{K}_j} \beta_{j,j^*}(X_{j^*} - \mu_{j^*}), \sigma_j^2\Big), \tag{4}$$

mit $\mathbf{K}_j = \{j^*|X_{j^*} \in \mathrm{pa}_{\mathcal{G}}(X_j)\}$, wobei μ_j der nicht bedingte Erwartungswert und σ_j^2 die nicht bedingte Varianz von X_j sind und β_{j,j^*} die reellen Koeffizienten darstellen, welche den Einfluss von X_{j^*} auf X_j beschreiben, so wird $(p, \mathcal{G})$ als Gaußsches Bayessches Netzwerk (GBN) bezeichnet (Geiger und Heckerman, 1994).

Die Analyse eines Datensatzes mit Hilfe von GBN liefert eine Schätzung der Netzwerkstruktur $\mathcal{G}$. Dies geschieht, indem die Likelihood betrachteter Kandidaten-Graphen maximiert wird, welche im Folgenden, in vereinfachter Form, hergeleitet wird. Das Vorgehen orientiert sich an Geiger und Heckerman (1994) sowie an Ickstadt et al. (2011). Die in Gleichung (4) beschriebene Verteilung der X_j ist eine Linearkombination normalverteilter Zufallsvariablen, die gemäß den Eigenschaften der Normalverteilung gemeinsam wieder normalverteilt sind. Parametrisiert wird diese durch den Erwartungsvektor $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_d)'$ und die Präzisionsmatrix $\boldsymbol{M}$, die Inverse der Kovarianzmatrix. Gemäß der Arbeit von Shachter und Kenley (1989) lassen sich $\boldsymbol{\mu}$ und $\boldsymbol{M}$ aus dem System der bedingten Wahrscheinlichkeiten $(p, \mathcal{G})$ herleiten. Damit stellen die Parameter $\boldsymbol{\mu}$, $\boldsymbol{\sigma} = (\sigma_1^2, \ldots, \sigma_d^2)'$ und

$\boldsymbol{B} = (\boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_d)$, mit $\boldsymbol{\beta}_j = (\beta_{j,1}, \ldots, \beta_{j,j-1})$, welche ein Gaußsches Bayessches Netzwerk beschreiben, eine alternative Parametrisierung der multivariaten Normalverteilung dar. Das Ziel der GBN-Analyse ist die Aufdeckung der Netzwerkstruktur $\mathcal{G}$, die übrigen Parameter sind im Regelfall von geringem Interesse und werden mittels Integration eliminiert. Durch die Verwendung konjugierter a priori Verteilungen kann die Berechnung sehr effizient umgesetzt werden. Ist die betrachtete Verteilung multivariat normal, fällt die Wahl der a priori Verteilung auf die Normal-Wishart Verteilung. Diese ist von der Form $p(\boldsymbol{\mu}|\boldsymbol{M})p(\boldsymbol{M})$, wobei $p(\boldsymbol{\mu}|\boldsymbol{M})$ eine multivariate Normalverteilung ist und $p(\boldsymbol{M})$ die Wishart Verteilung repräsentiert. Die Verteilung der letzteren kann ebenfalls mit den Parametern $\boldsymbol{\sigma}$ und $\boldsymbol{B}$ beschrieben werden. Zudem lässt sich die Wishart Verteilung faktorisieren, so dass

$$p(\boldsymbol{M}) = p(\boldsymbol{\sigma}, \boldsymbol{B}) = \prod_{j=1}^{d} p(\sigma_j^2, \boldsymbol{\beta}_j)$$

gilt (Geiger und Heckerman, 1994). Somit lautet für ein gegebenes $\mathcal{G}$ die Likelihood einer Stichprobe $\mathcal{X} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ unabhängig identisch multivariat normal verteilter Zufallsvariablen $X_1, \ldots, X_d$

$$\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{M}_{\mathcal{G}}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \prod_{i=1}^{n} p(\boldsymbol{x}_i|\boldsymbol{\mu}, \boldsymbol{M}_{\mathcal{G}}) \ , \tag{5}$$

wobei $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{id})'$ und $\boldsymbol{M}_{\mathcal{G}}$ so gewählt wird, dass die Struktur der bedingten Unabhängigkeiten aus $\mathcal{G}$ gewahrt bleibt.

Wird $\boldsymbol{\mu}$ ausintegriert $\left(\int \mathcal{L}(\boldsymbol{\mu}, \boldsymbol{M}_{\mathcal{G}}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)p(\boldsymbol{\mu}|\boldsymbol{M}_{\mathcal{G}})p(\boldsymbol{M}_{\mathcal{G}})d\boldsymbol{\mu}\right)$, erhält man einen vereinfachten Ausdruck, welcher wiederum unter Ausnutzung der Faktorisierbarkeit (Geiger und Heckerman, 1994) geschrieben werden kann als

$$\mathcal{L}(\boldsymbol{M}_{\mathcal{G}}|\mathcal{X}) = \mathcal{L}(\boldsymbol{\sigma}, \boldsymbol{B})|\mathcal{X}) = \prod_{j=1}^{d} \mathcal{L}\left(\sigma_j^2, \boldsymbol{\beta}_j|\mathcal{X}^{(j \cup \mathbf{K}_j)}\right) \ ,$$

wobei $\mathcal{X}^{(\mathcal{J})}$ die Spalten von $\mathcal{X}$ mit Indizes in $\mathcal{J}$ beschreibt. Dieser lässt sich weiter zu der marginalen Likelihood von $\mathcal{G}$ umformen

$$\begin{aligned} \mathcal{L}(\mathcal{G}|\mathcal{X}) &= \int \mathcal{L}\left(\boldsymbol{\sigma}, \boldsymbol{B}|\mathcal{X}\right) p(\boldsymbol{\sigma}, \boldsymbol{B}) d\boldsymbol{\sigma} d\boldsymbol{B} \\ &= \prod_{j=1}^{d} \int \mathcal{L}\Big(\sigma_j^2, \boldsymbol{\beta}_j|\mathcal{X}^{(\{j\} \cup \mathbf{K}_j)}\Big) p(\sigma_j^2, \boldsymbol{\beta}_j) d\sigma_j d\boldsymbol{\beta}_j \ . \end{aligned} \tag{6}$$

Der in dieser Arbeit verfolgte Schätzvorgang ist in ein Markov Chain Monte Carlo (MCMC) Gerüst eingebettet. Der Raum der möglichen Graphen wird unter Verwendung des von Madigan und York (1995) beschriebenen Verfahrens durchsucht, welches auf den sogenannten Kantenoperationen basiert. Dahinter stehen drei Möglichkeiten, einen Graphen (DAG) an einer Stelle zu verändern, und zwar indem zwischen zwei Knoten eine fehlende Kante hinzugefügt wird, eine vorhandene entfernt oder eine vorhandene umgedreht wird.

2.7. Diskussion

Ein generelles Ranking der vorgestellten Methodengruppen ist schwierig. In jeder der Gruppen findet sich ein Verfahren, welches bei sorgfältiger Anwendung mit entsprechender problemspezifischer Anpassung gute Ergebnisse liefert. Dies hat sich bei dem dritten Wettbewerb zu Netzwerkinferenz (DREAM3, Dialogue for Reverse Engineering Assessment and Methods, www.the-dream-project.org) gezeigt. Die Netzwerkinferenzmethoden, welche die fünf bestplatzierten Teams verwendet haben, gehörten unterschiedlichen Gruppen an (Stolovitzky et al., 2007). In die gleiche Richtung deuten auch aktuelle Publikationen hin, in welchen Netzwerkinferenzmethoden verglichen werden, zum Beispiel Villaverde et al. (2013). Es gibt kein global überlegenes Verfahren. Für jedes neue Problem, für jede neue Datenkonstellation wird empfohlen mehrere Alternativen zu betrachten. In den Arbeiten von Marbach et al. (2010), de Smet und Marchal (2010) oder Altay und Emmert-Streib (2010), welche aktuelle Netzwerkinferenzmethoden hinsichtlich ihrer Leistungsfähigkeit bei unterschiedlichen Problemstellungen systematisch vergleichen, wird sogar vorgeschlagen mehrere Methoden parallel anzuwenden, um eine „community prediction“ zu erhalten. Das „community“ Konzept sieht vor unterschiedliche Schätzer beziehungsweise einen Schätzer unter verschiedenen Konfigurationen auf einen Datensatz anzuwenden, um auf diese Weise eventuelle Schwächen einzelner auszugleichen und so eine bessere Schätzung zu erhalten.

Im Allgemeinen gilt für die im Abschnitt 2.5 vorgestellten Petri-Netze, dass sie bei der Bestimmung der Parameter in einem Netzwerk gute Ergebnisse zeigen. Zudem ermöglicht ihre Struktur, welche neben den Kanten aus zwei Arten von Knoten besteht, zusätzliche Analysemöglichkeiten, die die anderen Inferenzmethoden in diesem Umfang nicht bieten

(Küffner et al., 2010). Durch die Manipulation der Transitionen können beispielsweise äußere Einflüsse auf das System untersucht werden. Ein großer Nachteil des Verfahrens ist aber, dass zur Bestimmung der Anfangspopulation viel Expertenwissen eingebracht werden muss. Eine Initiierung mit einer Zufallspopulation ist zwar möglich, führt aber oft in eine Sackgasse. Vor allem bei Netzwerken mit vielen Knoten kann eine nicht zutreffende Population zu einem beträchtlichen Anstieg der Rechenzeit führen.

Für die auf Korrelation bzw. auf Regressionsverfahren basierenden Methoden aus Abschnitt 2.3 gilt nach Krämer et al. (2009), dass die Ridge-Regression im Vergleich zu den Lasso-Verfahren eher konservativ ist. Dies bedeutet, dass erstere bei Netzwerken mit vielen Verbindungen dazu tendiert viele falsch negative Kanten vorzuschlagen. Umgekehrt erzeugen die Lasso-Verfahren in Netzwerken mit wenigen Verbindungen tendenziell vermehrt falsch positive Kanten.

Ansätze, welche auf der mutual information aufbauen (vergleiche Abschnitt 2.4), haben den Vorteil, dass keine Annahmen über die zu Grunde liegende Verteilung der Knoten getroffen werden, wie das beispielsweise bei den GBNs oder den Regressionsmodellen der Fall ist. Ferner sind sie in den meisten Fällen skalierbar, was bedeutet, dass sie auch auf große Netzwerke mit vielen Knoten bei moderaten Rechenkosten anwendbar sind (Villaverde et al., 2013). Ein Nachteil dieses Ansatzes könnte jedoch die Eigenschaft sein, dass die mutual information immer für Paare berechnet wird und eine Interaktion, welche zwischen mehreren Knoten besteht, unter Umständen nicht erkannt wird. Zudem erlaubt dieser Ansatz keine Prädikationen.

Die Gruppe der Verfahren, welche primär die Kovarianzmatrix nutzen (Abschnitt 2.2), ist vorwiegend für die Auswertung von Microarry Experimenten konzipiert worden. Diese Verfahren sind in der Lage effizient mit großen Datenmengen umzugehen. Sie sind vor allem bei der Analyse von Fragestellungen, in denen die Anzahl der Knoten die Anzahl der Beobachtungen deutlich überschreitet, eine gute Wahl. Problematisch ist aber, dass tendenziell sehr volle Netzwerke gefunden werden, die Methode also anfällig für viele falsch positive Kanten ist.

Für die in dieser Arbeit betrachtete Fragestellung hat das Konzept der Bayesschen Netzwerke die größten Vorteile. Diese manifestieren sich vor allem im Hinblick auf die Problematik der vermischten Daten. Das Konzept bietet die Möglichkeit Vorwissen einzubeziehen, kann aber auch mittels nicht informativer a priori Verteilungen ohne dieses auskommen. Dies gilt sowohl für die Netzwerkstruktur als auch für die im weiteren Verlauf behandelte Clusterstruktur der Beobachtungen. Günstig ist auch die Eigenschaft als Bayessches Verfahren, selbst mit kleinen Stichproben zuverlässig zu arbeiten, gleichzeitig aber auch große Datenmengen effizient nutzen können. Ferner bietet die Markov Chain Monte Carlo gestützte Netzwerkschätzung Schutz vor Überanpassung und hat den Vorzug, dass die Wahrscheinlichkeit jeder einzelnen Kante überprüft werden kann (vergleiche Abschnitt 3.3.1). Dies ermöglicht es auch prädiktive Aussagen zu treffen. Keines der anderen Verfahren bietet diese Vorteile.

3. Entwickelte und adaptierte Methoden für Netzwerkinferenz

In diesem Abschnitt wird das zentrale Element der vorliegenden Arbeit, die Entwicklung eines Konzepts zur Entmischung mittels nichtparametrischer Bayesscher Netzwerke, vorgestellt. Die in diesem Rahmen durchgeführten Vorarbeiten konnten bereits erfolgreich publiziert und öffentlich diskutiert werden (Ickstadt et al., 2011). Viele der erhaltenen Anregungen sind im vorliegenden Text aufgegriffen worden.

Nach einer kurzen Einführung in das grundlegende Konzept der Bayesschen Netzwerke im Abschnitt 2.6 wird nun deren Erweiterung, die nichtparametrischen Bayesschen Netzwerke (NPBN) besprochen. Dazu gehört auch die Betrachtung zweier stochastischer Prozesse, des Dirichlet-Prozesses und des Pitman-Yor-Prozesses, die viel zu der Leistungsfähigkeit des Verfahrens beisteuern. Am Ende des Kapitels wird darüber hinaus auf die Nachbereitung der Schätzergebnisse eingegangen.

3.1. Zufällige Wahrscheinlichkeitsmaße

Ein für das NPBN-Konzept essenzieller Baustein sind die zufälligen Wahrscheinlichkeitsmaße (Kingman, 1967, 1978). Diese werden im Abschnitt 3.2 für die Gewichte der Mischverteilung genutzt (Gleichung (12)) sowie für diverse Berechnungen innerhalb der MCMC-Iterationen. Die folgende Darstellung orientiert sich an der Arbeit von Hjort et al. (2010), die wiederum der von Kingman (1967) eingeführten und später ausgebauten Definition folgt.

Sei $\mathcal{Z}^{(\infty)} = (\mathcal{Z}_n)_{n\geq 1}$ eine Folge von Zufallsvariablen definiert über dem Wahrscheinlichkeitsraum $(\Omega, \mathscr{F}, \mathbb{P})$. Die $\mathcal{Z}_i$ nehmen Werte aus einem separablen metrischen Raum $\mathbb{X}$ ausgestattet mit der Borel-σ-Algebra $\mathscr{X}$ an. Ferner sei $Z^{(\infty)}$ austauschbar (exchangeable), d.h. für jedes $n \geq 1$ und jede Permutation π der Indizes $1, \ldots, n$ stimmt die Wahrscheinlichkeitsverteilung des Zufallsvektors $(\mathcal{Z}_1, \ldots, \mathcal{Z}_n)$ mit der Wahrscheinlichkeitsverteilung von $(\mathcal{Z}_{\pi(1)}, \ldots, \mathcal{Z}_{\pi(n)})$ überein.

Weiter sei $\mathcal{M}_{\mathbb{X}}$ der Raum der beschränkten endlichen Maße auf $(\mathbb{X}, \mathscr{X})$, so dass für jedes μ aus $\mathcal{M}_{\mathbb{X}}$ und jede beschränkte Menge A aus $\mathscr{X}$ gilt $\mu(A) < \infty$. $\mathscr{M}_{\mathbb{X}}$ steht für die Borel-σ-Algebra auf $\mathcal{M}_{\mathbb{X}}$. Sei $\tilde{\mu}$ eine messbare Abbildung von $(\Omega, \mathscr{F}, \mathbb{P})$ nach $(\mathcal{M}_{\mathbb{X}}, \mathscr{M}_{\mathbb{X}})$, so dass für jegliche $A_1, \ldots, A_s$ $(s \in \mathbb{N})$ in $\mathscr{X}$, mit $A_{s'} \cap A_{s^*} = \emptyset$ für alle $s' \neq s^*$, die zufälligen Maße $\tilde{\mu}(A_1), \ldots, \tilde{\mu}(A_s)$ gemeinsam unabhängig sind. Gilt zusätzlich $0 < \tilde{\mu}(\mathbb{X}) < \infty$ fast sicher, dann heißt $\tilde{p} = \tilde{\mu}/\tilde{\mu}(\mathbb{X})$ zufälliges Wahrscheinlichkeitsmaß (Hjort et al., 2010, Definition 3.19).

Eine austauschbare Folge von Zufallsvariablen $Z^{(\infty)}$ wird gewöhnlich über folgendes Modell unter Verwendung von bedingter Unabhängigkeit und Gleichheit in Verteilung beschrieben,

$$\begin{aligned} \mathcal{Z}_i \mid \tilde{p} &\overset{\text{i.i.d.}}{\sim} \tilde{p}\,, \quad i \geq 1 \\ \tilde{p} &\sim Q\,, \end{aligned} \tag{7}$$

vergleiche Hjort et al. (2010). Dabei ist $\tilde{p}$ ist ein zufälliges Wahrscheinlichkeitsmaß definiert auf $(\Omega, \mathscr{F}, \mathbb{P})$ mit Werten aus der Potenzmenge von $\mathbb{X}$ und Q ist seine Verteilung. Das Produkt $\tilde{p}^{\,n} = \prod_{i=1}^{n} \tilde{p}$ steht somit für die bedingte Wahrscheinlichkeitsverteilung von $(\mathcal{Z}_1, \ldots, \mathcal{Z}_n)$, gegeben $\tilde{p}$. Aus Bayesscher Perspektive lässt sich Q als a priori Verteilung für die beschriebene zufällige Wahrscheinlichkeitsverteilung interpretieren.

Für die Analyse von Partitionsstrukturen und Mischverteilungen eignen sich diskrete zufällige Wahrscheinlichkeitsmaße besonders gut. Sie sollen daher im Folgendem genauer betrachtet werden.

3.1.1. Wahrscheinlichkeitsfunktion austauschbarer Partitionen

Im Folgenden werden die Wahrscheinlichkeiten für das Eintreten bestimmter Partitionen vorgestellt.

Seien $\mathcal{Z}_1, \ldots, \mathcal{Z}_n$ Zufallsvariablen, generiert gemäß Modell (7) mit diskretem Wahrscheinlichkeitsmaß $\tilde{p}$. Durch die Diskretheit ist das Auftreten von Bindungen, also $\mathbb{P}(\mathcal{Z}_i = \mathcal{Z}_{i^*}) > 0$ für $i \neq i^*$, möglich. Sei dementsprechend Ψ_n eine zufällige Partition der Zahlen $\{1, \ldots, n\}$ mit $C_* \in \Psi_n$: $i \wedge i^* \in C_* \Leftrightarrow \mathcal{Z}_i = \mathcal{Z}_{i^*}$. Sei weiter $k \in \{1, \ldots, n\}$ und $\{C_1, \ldots, C_k\}$ eine Partition von $\{1, \ldots, n\}$ in k Mengen C_i. Damit ist $\{C_1, \ldots, C_k\}$ eine

mögliche Realisierung von Ψ_n, die angibt, wie häufig unter den $\mathcal{Z}_1, \ldots, \mathcal{Z}_n$ eine bestimmte Ausprägung vorkommt. C_1 enthält dabei die Indizes der $\mathcal{Z}_i$, welche die erste der k unterschiedlichen Ausprägungen angenommen haben, C_2 die der zweiten und so weiter. Weiter sei mit $n_i :=$ Kardinalität(C_i) $\forall i$ und $(n_1, \ldots, n_k) \in \Delta_{n,k}$

$$\Delta_{n,k} = \left\{ (n_1, \ldots, n_k) : n_i \geq 1, \sum_{i=1}^{k} n_i = n \right\}$$

die Menge der Möglichkeiten, $n \in \mathbb{N}$ in k positive Summanden zu zerlegen, definiert. Dann gilt, dass die Wahrscheinlichkeit in der Menge $\{\mathcal{Z}_1, \ldots, \mathcal{Z}_n\}$ k verschiedene Ausprägungen mit den absoluten Häufigkeiten $C_1, \ldots, C_k$ zu beobachten gleich der Wahrscheinlichkeit ist, dass n in $n_1, \ldots n_k$ Summanden zerfällt,

$$\mathbb{P}\big[\Psi_n = \{C_1, \ldots C_k\}\big] = \Pi_k^{(n)}(n_1, \ldots, n_k) \ .$$

$\Pi_k^{(n)}$ heißt Wahrscheinlichkeitsfunktion austauschbarer Partitionen und wird nach der englischen Benennung exchangeable partition probability function mit EPPF abgekürzt (Pitman, 1995). $\Pi_k^{(n)}$ erlaubt es prädiktive Aussagen über die Form der C_i zu treffen, was sich auf die Clusterstruktur der $\mathcal{Z}_i$ übertragen lässt.

Sei $\Pi_k^{(n)}$, $(1 \leq k \leq n,\ n \geq 1)$ die EPPF einer austauschbaren Folge von Zufallsvariablen. Weiter sei $\boldsymbol{Z} = (\mathcal{Z}_1, \ldots, \mathcal{Z}_n)$ der Vektor der realisierten Stichprobenvariablen, wobei k unterschiedliche Ausprägungen $\mathcal{Z}_1^*, \ldots, \mathcal{Z}_k^*$ angenommen wurden, dann trägt eine neue Stichprobenvariable $\mathcal{Z}_{n+1}$ eine bislang nicht vorkommende Ausprägung $\mathcal{Z}^*$ mit Wahrscheinlichkeit

$$\mathbb{P}\big[\mathcal{Z}_{n+1} = \mathcal{Z}^* | \boldsymbol{Z}\big] = \frac{\Pi_{k+1}^{n+1}(n_1, \ldots, n_k, 1)}{\Pi_k^{(n+1)}(n_1, \ldots, n_k)} \quad , \quad \mathcal{Z}^* \notin \{\mathcal{Z}_1^*, \ldots, \mathcal{Z}_k^*\} \ .$$

Ferner ist die Wahrscheinlichkeit, dass eine neue Beobachtung eine bereits vorkommende Ausprägung $\mathcal{Z}_i^*$ trägt, von der Form

$$\mathbb{P}\big[\mathcal{Z}_{n+1} = \mathcal{Z}_i^* | \boldsymbol{Z}\big] = \frac{\Pi_k^{n+1}(n_1, \ldots, n_i + 1, \ldots, n_k)}{\Pi_k^{(n)}(n_1, \ldots, n_k)} \quad , \quad \mathcal{Z}_i^* \in \{\mathcal{Z}_1^*, \ldots, \mathcal{Z}_k^*\}$$

vergleiche Hjort et al. (2010).

3.1.2. Dirichlet-Prozess

Der Dirichlet-Prozess steht für eine ganze Klasse von diskreten zufälligen Wahrscheinlichkeitsverteilungen (Ferguson, 1973). Er zeichnet sich unter anderem dadurch aus, dass er eine Partitionsstruktur erzeugen kann, die theoretisch unbeschränkt ist, d.h. unendlich viele Partitionen enthalten kann. Diese Eigenschaft ermöglicht es auch eine Mischung aus theoretisch unendlich vielen Verteilungen im Modell zu handhaben. Sie macht den nichtparametrischen Teil von NPBN aus.

Formal stellt ein Dirichlet-Prozess ein zufälliges Wahrscheinlichkeitsmaß $\tilde{p}$ auf $\mathscr{X}$ dar, so dass für jede endliche Partition $(B_1, \ldots, B_k)$ von $\mathscr{X}$ gilt

$$\tilde{p}(B_1), \ldots, \tilde{p}(B_k) \sim \mathrm{Dir}\Big(\theta P_0(B_1), \ldots, \theta P_0(B_k)\Big).$$

Dabei ist P_0 ein Maß auf $\mathscr{X}$, das sogenannte Grundmaß, $\theta \in \mathbb{R}$ der sogenannte Präzisionsparameter und Dir die Dirichletverteilung.

Der Dirichlet-Prozess lässt sich auf verschiedene Weisen konstruieren. Dies ist sowohl für theoretische Betrachtungen als auch für die Praxis nützlich, da in jeder Konstruktionsvorschrift unterschiedliche Qualitäten sichtbar werden.

Eine sehr anschauliche Konstruktionsvorschrift greift auf die sogenannte Pólya Urne zurück. Betrachtet wird eine Urne, die unendlich viele Kugeln fassen kann. Es werden θ Kugeln der Farbe 1 hinein getan. Dann wird mehrfach hintereinander eine Kugel entnommen. Wenn bei einer Ziehung eine Kugel der Farbe 1 gezogen wird, wird diese zurückgelegt, zusammen mit einer weiteren Kugel einer Farbe, die sich noch nicht in der Urne befindet. Wird eine Kugel der Farbe $\neq 1$ gezogen, so wird diese zusammen mit einer weiteren der gleichen Farbe zurückgelegt. Bei der n-ten Ziehung gilt, dass die Farbe der gezogenen Kugel $\mathbf{U}_n$

$$\mathbf{U}_n | \mathbf{U}_1, \ldots, \mathbf{U}_{n-1} = \begin{cases} F, \text{mit Wsk. } \frac{n_F}{\theta+n-1} \\ 1 \text{ , mit Wsk. } \frac{\theta}{\theta+n-1} \end{cases}$$

ist. Dabei ist $F \in \{2, \ldots, k\}$, n_F die Anzahl der Kugeln der Farbe F die sich bereits in der Urne befinden und k steht für die Anzahl der unterschiedlichen Farben in der Urne. Wenn die Farben aus dem Grundmaß P_0 gezogen werden, so ist die Folge $\mathbf{U}_n$ eine Realisierung

eines Dirichlet-Prozesses mit den Parametern θ und P_0 und die relativen Häufigkeiten der Farben ergeben, für jedes feste n, ein zufälliges Wahrscheinlichkeitsmaß. Die Reihenfolge der gezogenen Kugeln ist dabei austauschbar. Auf den allgemeinen Fall aus Abschnitt 3.1.1 bezogen entsprechen die $\mathbf{U}_n$ den $\mathcal{Z}_n$ und die F den $\mathcal{Z}^*$. Diese Konstruktionsvorschrift und die daraus abgeleiteten Eigenschaften werden in der Arbeit von Blackwell und MacQueen (1973) ausführlich diskutiert.

Eine zweite sehr verbreitete Möglichkeit einen Dirichlet-Prozess zu erzeugen, ist die sogenannte stick-breaking Konstruktion. Diese Form wird auch als die Sethuraman Repräsentation bezeichnet (Sethuraman, 1994).

Sei $(\mathcal{E}_i)_{i\geq 1}$ eine Folge unabhängiger Zufallsvariablen mit Werten aus $[0,1]$,

$$\tilde{p}_1 = \mathcal{E}_1, \quad \tilde{p}_i = \mathcal{E}_i \prod_{j=1}^{i-1}(1-\mathcal{E}_j) \quad i \geq 2 \ . \tag{8}$$

Sind die $\mathcal{E}_i$ unabhängig identisch Beta$(1,\theta)$ verteilt, so ist $\sum_{i=1}^{\infty} \tilde{p}_i \delta_{\theta_i}$ mit $\theta_i \sim P_0$ ein Dirichlet-Prozess mit Parametern θ und P_0.
Die im folgendem Kapitel zentrale EPPF des Dirichlet-Prozesses lautet dann

$$\Pi_k^{(n)}(n_1,\ldots,n_k) = \frac{\theta^k \cdot \Gamma(\theta)}{\Gamma(\theta+n)} \prod_{j=1}^{k} \Gamma(n_j) \ . \tag{9}$$

Eine weitere bemerkenswerte Eigenschaft des Dirichlet-Prozesses ist, dass die Wahrscheinlichkeit, dass eine neue Beobachtung zu keiner der bereits vorhandenen Klassen gehört, nur von n abhängt und mit jeder Ziehung sinkt. Korwar und Hollander (1973) haben gezeigt, dass für die Anzahl K_n der unterschiedlichen Gruppen beim Dirichlet-Prozess $K_n \underset{n\to\infty}{\longrightarrow} \theta \log(n)$ gilt.

3.1.3. Pitman-Yor-Prozess

Die Namensgebung im Fall des Pitman-Yor-Prozesses ist nicht eindeutig. Er ist auch als zwei Parameter Poisson-Dirichlet-Prozess (Pitman, 2003) oder Perman–Pitman–Yor-Prozess (Perman et al., 1992) bekannt. Für den Pitman-Yor-Prozess existiert ebenfalls eine stick-breaking Konstruktionsvorschrift (Pitman, 1995). Die Konstruktion verläuft analog

zu der in Gleichung (8) für den Dirichlet-Prozess beschriebenen, mit dem Unterschied, dass $\mathcal{E}_i \sim \text{Beta}(1-\sigma, \theta + i\sigma)$ mit $0 \leq \theta < 1$ und $\theta > -\sigma$ für $i \geq 1$, $i \in \mathbb{N}$.

Die EPPF des Pitman-Yor-Prozesses mit den Parametern θ und σ, also die Wahrscheinlichkeit, dass eine Menge der Größe n zu einer Partition aus k Mengen der Größen $n_1, \ldots, n_k$ zerfällt, ist nach Hjort et al. (2010) gegeben durch

$$\Pi_k^{(n)}(n_1, \ldots, n_k) = \frac{\prod_{i=1}^{k-1}(\theta + i\sigma) \cdot \Gamma(\theta + 1)}{\Gamma(\theta + n)} \prod_{j=1}^{k} \frac{\Gamma(n_j - \sigma)}{\Gamma(1 - \sigma)} \; . \tag{10}$$

Der Pitman-Yor-Prozess stellt eine Verallgemeinerung des Dirichlet-Prozesses dar. Hier wird, skaliert durch einen weiteren Parameter σ, die Anzahl der bereits beobachteten unterschiedlichen Ausprägungen $\mathcal{Z}_1^*, \ldots, \mathcal{Z}_k^*$ berücksichtigt und fließt in die Wahrscheinlichkeit der Ziehung einer neuen Ausprägung $\mathcal{Z}_{k+1}^*$ ein. Für die Anzahl der Gruppen K_n in Abhängigkeit von der Stichprobengröße gilt $K_n \underset{n\to\infty}{\longrightarrow} S_{\sigma,\theta} \cdot n^{\sigma}$, wobei $S_{\sigma,\theta}$ eine Zufallsvariable auf $\mathbb{R}^+$ ist (Pitman, 2003). Es ist bekannt, dass der Pitman-Yor-Prozess dem power law folgt. Das bedeutet, es werden nur wenige hohe und viele kleine Wahrscheinlichkeiten ($\mathcal{E}_i$) erzeugt. Wird der Pitman-Yor-Prozess in Zusammenhang mit Clustern verwendet, sind einige große Cluster und viele kleine zu erwarten.

3.2. Nichtparametrische Bayessche Netzwerke

Die im vorherigen Abschnitt vorgestellten Gaußschen Bayesschen Netzwerke werden im Rahmen des in dieser Arbeit behandelten Projekts zu nichtparametrischen Bayesschen Netzwerken weiterentwickelt (Ickstadt et al., 2011) und zu einem Ansatz für Netzwerkinferenz und Entmischung von vermischten Stichproben („Unmixing via Nonparametric Bayesian Networks“, UNPBN) ausgebaut (Wieczorek et al., 2015). Das Konzept des UNPBN wurde vom Verfasser der vorliegenden Arbeit gemeinsam mit den Autoren von Ickstadt et al. (2011) entwickelt. Die Validierung sowie die Erweiterung der Methoden um den Pitman-Yor-Prozess stellt eine eigenständige Leistung dar.

Die Kernidee ist, die Flexibilität von Mischverteilungen (mixture models) zu nutzen, um den Datensatz $\mathcal{X}$ als Gefüge verschiedener Netzwerke zu betrachten und für die einzelnen

Teile GBNs zu schätzen. Die Mischung erfolgt unter Berücksichtigung aller relevanten Parameter $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}, \mathcal{G})$. Das Modell für die Daten kann formuliert werden als

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}, \mathcal{G}) dP(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}, \mathcal{G}) \ ,$$

wobei wie in Abschnitt 2.6 $\boldsymbol{\mu}$ und $\boldsymbol{\sigma}$ die Vektoren der nicht bedingten Erwartungswerte $(\mu_1, \ldots, \mu_d)'$ und der Varianz $(\sigma_1^2, \ldots, \sigma_d^2)'$ darstellen und $\boldsymbol{B} = (\boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_d)$, $\boldsymbol{\beta}_j = (\beta_{j,1}, \ldots \beta_{j,j-1})$, vergleiche Seite 17. Die Mischverteilung P ist verteilt gemäß $\tilde{p}$, einer zufälligen Wahrscheinlichkeitsverteilung und $p(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}, \mathcal{G})$ ist eine multivariate Normalverteilung mit einer bedingten Unabhängigkeitsstruktur kompatibel zu $\mathcal{G}$. Für die Schätzung ist die auf $\mathcal{G}$ bedingte Gleichung

$$p(\boldsymbol{x}|\mathcal{G}) = \int p(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}, \mathcal{G}) dP(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{B}) \tag{11}$$

nützlich. Die diskrete Natur von P mit Träger $\boldsymbol{\mu}_h, \boldsymbol{\sigma}_h, \boldsymbol{B}_h$ und Wahrscheinlichkeiten w_h erlaubt es, die Mischung als Summe zu formulieren:

$$p(\boldsymbol{x}) = \sum_{h=1}^{N} w_h \ p(\boldsymbol{x}|\boldsymbol{\mu}_h, \boldsymbol{\sigma}_h, \boldsymbol{B}_h, \mathcal{G}). \tag{12}$$

Die a priori Verteilung der Mischungsgewichte w_h wird durch $\tilde{p}$ zugeordnet. Die a priori Verteilung für $\boldsymbol{\mu}_h, \boldsymbol{\sigma}_h, \boldsymbol{B}_h, \mathcal{G}$ ist für alle h gegeben durch das Grundmaß P_0 gemäß $\tilde{p}$. Die N verschiedenen Mischungskomponenten h (im Folgenden auch als Komponenten abgekürzt) können als Cluster bzw. Subgruppen im Datensatz interpretiert werden.

Die Zuordnung jedes Datenpunktes zu der entsprechenden Komponente wird durch den Allokationsvektor (Nobile und Fearnside, 2007) dokumentiert (vergleiche auch 2.1). Dieser hat die Form $\boldsymbol{l} = (l_1, \ldots, l_i, \ldots, l_n)'$ wobei $l_i \in \{1, \ldots, h, \ldots, N\}$.

Die Netzwerkstruktur $\mathcal{G}$ und der Allokationsvektor $\boldsymbol{l}$ stellen das primäre Ziel der NPBN-Analyse dar. Die verbleibenden Parameter $\boldsymbol{\mu}_h$, $\boldsymbol{\sigma}_h$ und $\boldsymbol{B}_h$ können aus der Likelihood herausintegriert werden. Dies führt mit $\boldsymbol{w} = (w_1, \ldots, w_N)$ zu

$$\mathcal{L}(\boldsymbol{w}, \boldsymbol{l}, \mathcal{G}|\mathcal{X}) = \prod_h \mathcal{L}(\mathcal{G}|\mathcal{X}_{(\mathcal{I}_h)}) \prod_h w_h^{n_h} \ . \tag{13}$$

Dabei ist $\mathcal{L}(\mathcal{G}|\mathcal{X}_{(\mathcal{I}_h)}) = \int L(\boldsymbol{\sigma}, \boldsymbol{B}|\mathcal{X}) p(\boldsymbol{\sigma}, \boldsymbol{B}) d\boldsymbol{\sigma} d\boldsymbol{B}$ (vergleiche Gleichung (6)), $\mathcal{X}_{(\mathcal{I})}$ beschreibt die Zeilen von $\mathcal{X}$ mit Indizes in $\mathcal{I}$, $\mathcal{I}_h = \{i \in \{1, \ldots, n\} | l_i = h\}$ und n_h gibt die

Kardinalität der Menge $\mathcal{I}_h$ an. Die Reduzierung der Parameterzahl hat den Vorteil, dass die Berechnung vereinfacht und dadurch beschleunigt wird. Ferner wird auf diesem Wege die Gefahr von Überanpassung (overfitting) gemindert, da der Schätzer Graphen begünstigt, die in der Breite, d.h. für mehrere Werte, eine hohe Likelihood liefern und nicht jene, die für eine einzelne Parametereinstellung optimal sind. Die marginale a posteriori Verteilung für $\boldsymbol{l}$ und $\mathcal{G}$ hat die Gestalt

$$p(\boldsymbol{l}, \mathcal{G}|\mathcal{X}) = \prod_h \mathcal{L}(\mathcal{G}|\mathcal{X}_{(\mathcal{I}_h)}) p_N(n_1, \ldots, n_N) p(\mathcal{G}) \ .$$

Dabei beschreibt $p_N(n_1, \ldots, n_N)$ die Wahrscheinlichkeit, dass die N verschiedenen Werte mit den absoluten Häufigkeiten $n_1, \ldots, n_N$ auftreten, welche durch die EPPF gegeben ist (vergleiche Abschnitt 3.1.1 sowie die Gleichungen 9 und 10). Die übergeordnete a posteriori Verteilung und damit das Ziel der MCMC-Analyse lautet

$$p(\boldsymbol{l}, \mathcal{G}, N|\mathcal{X}) = \prod_{h=1}^{N} \mathcal{L}(\mathcal{G}|\mathcal{X}_{(\mathcal{I}_h)}) p_N(n_1, \ldots, n_N) p(N) p(\mathcal{G}) \ , \tag{14}$$

wobei hier $p(N)$ die Verteilung der Anzahl der Komponenten darstellt.

Der MCMC-Teil des NPBN-Algorithmus umfasst zwei Arten von Schritten, den DAG Schritt (auch DAG move), welcher den DAG $\mathcal{G}$ aktualisiert sowie den Allokationsschritt (auch allocation move), welcher die Anzahl der Komponenten N und den Allokationsvektor $\boldsymbol{l}$ aktualisiert. Der DAG Schritt führt eine zufällige Kantenoperation (vergleiche Seite 8) durch, aus $\mathcal{G}$ entsteht ein neuer Graph $\mathcal{G}^v$, die Struktur von $\boldsymbol{l}$ bleibt unverändert. Die Vorschlagswahrscheinlichkeit für $\mathcal{G}^v$ ist gegeben durch $q(\mathcal{G}^v|\mathcal{G}) = \frac{1}{|M_\mathcal{G}|}$, $|M_\mathcal{G}|$ steht hierbei für die Anzahl der Nachbargraphen (vergleiche Seite 8) von $\mathcal{G}$. Die Akzeptanzwahrscheinlichkeit für $\mathcal{G}^v$ lautet

$$A(\mathcal{G}^v|\mathcal{G}) = \frac{p(\mathcal{G}^v|\mathcal{X})}{p(\mathcal{G}|\mathcal{X})} \cdot \frac{q(\mathcal{G}|\mathcal{G}^v)}{q(\mathcal{G}^v|\mathcal{G})} \ , \tag{15}$$

wobei $p(\mathcal{G}|\mathcal{X})$ aus Gleichung (14) hervorgeht.

Der Allokationsschritt wird aus Nobile und Fearnside (2007) adaptiert. Er umfasst fünf unterschiedliche Operationen (M1 Schritt, M2 Schritt, Teilungsschritt, Verschmelzungsschritt sowie Gibbs Schritt), von denen aber pro Iteration maximal einer ausgeführt wird. Die Graphenstruktur $\mathcal{G}$ bleibt dabei stets unverändert. Für eine detailliertere Beschreibung der einzelnen Schritte und des gesamten Ablaufs vergleiche Abschnitt C im Anhang.

Die Schritte M1 und M2 transferieren Beobachtungen zwischen zwei zufällig ausgewählten Komponenten. Der Unterschied zwischen ihnen besteht im Umfang und dem Verfahren zur Auswahl der Beobachtungen. Der Teilungsschritt (split move) sowie der Verschmelzungsschritt (merge move) verändern die Anzahl der Komponenten um jeweils eine Komponente. Bei ersterem werden zufällig ausgesuchte Beobachtungen aus einer Komponente in eine neuerzeugte versetzt. Bei dem zweiten wird aus zwei bislang getrennten Komponenten eine neue gebildet, welche alle Beobachtungen der beiden vorherigen enthält. Die Akzeptanzwahrscheinlichkeit für diese Schritte lautet

$$A(\boldsymbol{l}^v|\boldsymbol{l}) = \min\left\{1, \frac{p(\boldsymbol{l}^v, \mathcal{G}, N|\mathcal{X})}{p(\boldsymbol{l}, \mathcal{G}, N|\mathcal{X})} \frac{q(\boldsymbol{l}^v|\boldsymbol{l})}{q(\boldsymbol{l}|\boldsymbol{l}^v)}\right\} \ . \tag{16}$$

Dabei kann $p(\boldsymbol{l}^v, \mathcal{G}, N|\mathcal{X})$ aus Gleichung (14) entnommen werden, die Vorschlagswahrscheinlichkeit $q(.|.)$ weist zwischen den verschiedenen Schritten Unterschiede auf und muss gesondert betrachtet werden.

Die fünfte Operation des Allokationsschrittes stellt der sogenannte Gibbs Schritt dar. Hier wird eine einzelne Beobachtung $\mathbf{d}$ zufällig einer Komponente zugeordnet. Für alle möglichen Zuordnungen von $\mathbf{d}$ zu den Komponenten $1, \ldots, N$ wird die zugehörige, vollständig bedingte Verteilung aufgestellt. Der in die MCMC-Kette aufgenommene Vektor wird zufällig anhand dieser ermittelt. Weitere Erläuterungen sowie eine Beschreibung der Implementierung können im Anhang im Abschnitt C ab Seite 100 nachgelesen werden.

Das NPBN-Verfahren braucht a priori Verteilungen für $\boldsymbol{\mu}_h, \boldsymbol{\sigma}_h, \boldsymbol{B}_h, \mathcal{G}$ und für $w_1, \ldots, w_N$. Für $\mathcal{G}$ wird eine a priori Verteilung benutzt, welche über die Kardinalität der Elternmengen gleichverteilt ist (Friedman und Koller, 2003), für $\boldsymbol{\sigma}_h$ und $\boldsymbol{B}_h$ wird die Normal Wishart a priori Verteilung mit der Einheitsmatrix als Präzisionsmatrix und $d + 2$ Freiheitsgraden verwendet. Der Vektor der Erwartungswerte der multivariaten Normalverteilung $\boldsymbol{\mu}_h$ wird als Vektor von Nullen gewählt. Für N wird die Poisson-Verteilung mit Parameter $\lambda = 1$ gewählt und die w_h entstammen einem zufälligen Wahrscheinlichkeitsmaß. Für die a priori Verteilung des zufälligen Wahrscheinlichkeitsmaßes werden zwei Möglichkeiten berücksichtigt. Die erste stellt der Dirichlet-Prozess (Notation NPBN-DP) dar, da er sehr flexibel ist und die Form jeder beliebigen Multinomialverteilung annehmen kann. Dabei ist der Träger, d.h. die Anzahl der unterschiedlichen Kategorien, praktisch unbeschränkt.

Ferner zeichnet er sich durch ein umfassendes theoretisches Fundament und gute numerische Handhabbarkeit aus, vor allem in Verbindung mit MCMC-Methoden (vergleiche 3.1.2). Eine Alternative für die Wahl der a priori Verteilung ist der Pitman-Yor-Prozess (Notation NPBN-PY). Dieser stellt eine Verallgemeinerung des Dirichlet-Prozesses dar und besitzt die gleichen vorteilhaften Eigenschaften. Zusätzlich bietet er auch die Möglichkeit auf die Anzahl der gefundenen Komponenten Einfluss zu nehmen (vergleiche 3.1.3).

Damit setzt sich das vollständige Modell wie folgt zusammen. Die gemeinsame Verteilung der Knoten ist gegeben durch

$$P(X_1, \ldots, X_d) = \prod_{j=1}^{d} p\big(X_j | \mathrm{pa}_{\mathcal{G}}(X_j)\big) \quad ,$$

die Verteilung des einzelnen Knotens gegeben die Elternknoten lautet

$$X_j | \mathrm{pa}_{\mathcal{G}}(X_j) \sim N\Big(\mu_j + \sum_{\mathbf{K}_j} \beta_{j,j^*}(X_{j^*} - \mu_{j^*}), \sigma_j^2\Big)$$

und die diskrete Mischverteilung hat die Gestalt

$$p(X_1, \ldots, X_d) = \sum_{h=1}^{N} w_h \; p(X_1, \ldots, X_d | \boldsymbol{\mu}_h, \boldsymbol{\sigma}_h, \boldsymbol{B}_h, \mathcal{G})$$

mit der a priori Verteilung der Erwartungswerte und Kovarianzen

$$(\mu_j, M) \sim NW_d(\mathbf{0}, \mathbb{I}, d+2) \text{ mit } \mathbb{I} \text{ Einheitsmatrix,}$$

der a priori Verteilung der Graphenstrukur

$$\mathcal{G} \sim \prod_{j=1}^{d} \binom{d-1}{|pa_{\mathcal{G}(X_j)}|}^{-1} \quad ,$$

der a priori Verteilung der Mischungsgewichte

$$w \sim \mathrm{Dir}(1, \ldots, 1) \quad ,$$

falls der Dirichlet-Prozess verwendet wird beziehungsweise

$$w \sim \mathrm{PY}((1, \ldots, 1), \sigma) \quad ,$$

falls der Pitman-Yor-Prozess verwendet wird und der a priori Verteilung der Komponentenanzahl

$$N \sim Poi(1) \quad .$$

Für die in den folgenden Abschnitten präsentierten Berechnungen wird der Dirichlet-Prozess mit den Parametern $(\theta, \ldots, \theta)$, $\theta = 1$ initiiert. Für den Pitman-Yor-Prozess werden die Parametereinstellungen $\sigma = 0.2$ und $\theta = 1$ gewählt. Zu Vergleichszwecken werden auch weitere Einstellungen von σ untersucht, 0.5 und 0.8. In diesen Fällen wird darauf gesondert hingewiesen (Notation NPBN-PY 0.5 und NPBN-PY 0.8). Die Anzahl der Iterationen, die Ausdünnung sowie die Anzahl der Burn-In Iterationen werden in Anhängigkeit vom Datensatz festgelegt und im betreffenden Abschnitt angegeben. Eine Übersicht ist im Anhang in Tabelle 4 auf Seite 112 zu finden. Das beschriebene Verfahren wird in MATLAB (2009) implementiert.

3.3. Nachbereitung der Daten und abgeleitete Größen

Die Ausgabe der NPBN-Analyse besteht aus zwei MCMC-Ketten, also aus einer Menge von DAGs und einer Menge von Allokationsvektoren. Der Nachbereitungsschritt hat das Ziel, die in den MCMC-Ketten enthaltenen Informationen komprimiert zu extrahieren. Es ist zwar möglich, diese direkt zu verwenden, z.B. indem ein DAG aus der Menge als Repräsentant bestimmt wird. Die Qualität der Auswertung kann jedoch durch eine Nachbereitung der direkten Ausgabe deutlich verbessert werden. In englischsprachiger Literatur wird von postprocessing gesprochen. Auf diese Weise profitiert die Analyse von den Informationen, welche in der gesamten Stichprobe enthalten sind. Hierzu werden im Folgenden die in der vorliegenden Arbeit zum Einsatz gebrachten Verfahren zur Nachbereitung der Daten sowie zur Berechnung von Kennzahlen vorgestellt.

3.3.1. Nachbereitung der Graphenstruktur: Matrix der a posteriori Kantenwahrscheinlichkeiten

Die in der MCMC-Simulation gezogenen DAGs werden in Form von Adjazenzmatrizen gespeichert. Die darin enthaltenen Informationen über das untersuchte Netzwerk können in Form einer posterior edge probability matrix (*pep*-Matrix) zusammengefasst und visua-

lisiert werden. Sie ist von der Form $\mathbb{R} \cap [0,1]^{d \times d}$ und der Eintrag in der j-ten Zeile und j^*-ten Spalte (pep_{jj^*}) kann als Wahrscheinlichkeit für die Existenz einer gerichteten Kante vom j-ten Knoten zum j^*-ten Knoten interpretiert werden. Werte nahe eins sind ein starker Hinweis darauf, dass die Knoten verbunden sind. Werte nahe null sind ein starker Hinweis darauf, dass zwischen den Knoten keine Beziehung besteht. Werte im Bereich um 0.5 lassen sich nur schlecht interpretieren. Die Berechnung der *pep*-Matrix erfolgt komponentenweise als arithmetisches Mittel der Adjazenzmatrixeinträge (vergleiche Seite 9) aus der MCMC-Kette der DAGs

$$pep_{jj^*} = \sum_{s=1}^{r} \mathcal{A}^s_{jj^*}/r \ ,$$

wobei s der Index und r die Anzahl der betrachteten MCMC-Iterationen sind. Ist allein das Muster der Interaktionsstruktur und nicht die Bestimmung der Richtungen der Kanten das Ziel der Analyse, bietet es sich an, ungerichtete Netzwerke zu betrachten. Diese haben den Vorteil, dass sie deutlich robuster geschätzt werden können. Die Berechnung der *pep*-Matrix erfolgt genauso wie zuvor. Vorher müssen lediglich die Adjazenzmatrizen transformiert werden zu

$$\text{trans}(\mathcal{A}_{j^*j}) := \max(\mathcal{A}_{jj^*}, \mathcal{A}_{j^*j}) \quad , \quad j, j^* = 1, \ldots, d.$$

Die *pep*-Matrizen lassen sich direkt in eine graphische Repräsentation der geschätzten Netzwerke überführen. Alle Kanten, deren korrespondierende *pep*-Werte oberhalb einer Schranke liegen, gelten als mit hoher Wahrscheinlichkeit vorhanden und werden eingezeichnet. Alle Kanten, deren korrespondierende *pep*-Werte unterhalb einer zweiten Schranke liegen, gelten als mit hoher Wahrscheinlichkeit nicht vorhanden. Für Kanten, deren *pep*-Werte dazwischen liegen, kann keine Aussage getroffen werden. Um keine relevanten Kanten zu übersehen, wird die obere Schranke gleich 0.6 gewählt. Die Information über das Fehlen einer Kante ist genauso wichtig wie die über ihre Präsenz. Aus diesen Grund werden in dieser Arbeit die Intervalle symmetrisch gewählt und die untere Schranke wird gleich 0.4 gesetzt.

3.3.2. Nachbereitung der Allokationen: Reinheit (pco)

Während des Schätzvorgangs werden die Mengen, welche die zu den Komponenten zugehörige Beobachtungen enthalten, sowohl aufgespalten als auch zusammengeführt, so dass sich ihre Anzahl ändern kann. Aus diesen Grund ist eine durchgehende Benennung der Komponenten nicht praktikabel. Sie werden in jeder Iteration zufällig neu benannt. Bei Gegenüberstellung zweier Vektoren ist daher nicht erkennbar, ob eine Beobachtung einem anderen Cluster zugeordnet wurde oder ob das Cluster nur umbenannt wurde. Dieses Problem wird label switching genannt und ist die Ursache dafür, dass die Allokationsvektoren nicht direkt ausgewertet werden können. Es besteht zwar grundsätzlich die Möglichkeit einen einzelnen Allokationsvektor als Repräsentanten heranzuziehen. Dieses Vorgehen lässt aber den Großteil der in der MCMC-Kette enthaltenen Information ungenutzt. Eine alternative Möglichkeit, das label switching Problem zu lösen, bietet die Arbeit von Fritsch und Ickstadt (2009). Ihre Methode basiert auf der Maximierung des adjustierten Rand Indexes. Sie erlaubt es, die Allokationsvektoren einer MCMC-Kette zu einem einzigen Vektor zusammenzufassen. Dabei ist es möglich, die maximale Clusteranzahl vorzugeben oder diese automatisch bestimmen zu lassen. Dieses Verfahren ist in R (R Development Core Team, 2013) implementiert und über das Paket `mcclust` (Fritsch, 2009) zugänglich. Auf dieses Vorgehen wird bei der Berechnung des Silhouettenkoeffizienten für die Analyse der Komponentenzahl im Abschnitt 4.4 zurückgegriffen.

Bei der Bewertung der Leistung von NPBN bei der Entmischung soll aber direkt mit dessen Ausgabe, d.h. der Kette der Allokationsvektoren, gearbeitet werden. Daher erfolgt die Bewertung basierend auf der Reinheit/Homogenität innerhalb der Cluster. Damit ist gemeint, dass sich in einem Cluster im Idealfall nur Beobachtungen befinden sollten, die aus einer Komponente stammen. Für jedes Objekt $l_i^{s,h}$ im Allokationsvektor aus Iteration s wird für jede Komponente h die wahre Komponente durch Abgleich mit dem Versuchsaufbau bestimmt. Eine Beobachtung gilt als richtig zugeordnet (und die Indikatorfunktion $I(l_i^{s,h})$ wird auf 1 gesetzt), wenn sie aus der gleichen Komponente stammt wie die Mehrheit der Beobachtungen in ihrem Cluster. Alle Beobachtungen, die aus einer anderen Komponente stammen, gelten als falsch zugeordnet (und die Indikatorfunktion $I(l_i^{s,h})$ wird auf 0 gesetzt).

Diese Überprüfung findet komponentenweise für jede Beobachtung statt. Der Anteil der richtig zugeordneten Beobachtungen (*pco*) für eine NPBN-Schätzung lässt sich mit

$$pco = \frac{1}{r}\sum_{s=1}^{r}\frac{1}{N^s}\sum_{h=1}^{N^s}\frac{1}{n_h^s}\sum_{i=1}^{n_h^s} I(l_i^{s,h}) \cdot 100$$

berechnen, wobei r die Anzahl der betrachteten MCMC-Iterationen, n_h^s die Größe des Clusters h und N^s die geschätzte Anzahl der Cluster im Allokationsvektor in Iteration s ist. Durch die Multiplikation mit 100 wird der Quotient in eine prozentuale Angabe umgerechnet. Die Berechnung der *pco*-Werte für die Analyseergebnisse der Referenzverfahren (aus Kapitel 4) erfolgt analog.

3.3.3. Nachbereitung der Allokationen: Silhouettenkoeffizient

Die im Folgendem vorgestellte Kennzahl, der Silhouettenkoeffizient, im Englischen unter der Bezeichnung average silhouette width (ASW) bekannt, dient der Beurteilung der durch ein NPBN-Verfahren bestimmten Komponentenanzahl. Der ASW erfasst, wie dicht die Datenpunkte in einem Cluster liegen und wird häufig für die Bestimmung der optimalen Clusteranzahl herangezogen (Rousseeuw, 1987). Er bietet einen guten Anhaltspunkt zur Beurteilung der Qualität einer Clusterung und ist geeignet, um bei einem Vergleich von Clusterungsergebnissen, die mit dem gleichen Verfahren unter verschiedenen Parametern erzielt wurden, die richtige Einstellung zu finden. Obwohl an das Clusterverfahren selbst keine Anforderungen gestellt werden, ist der Vergleich verschiedener Clusterverfahren in der Anwendung nicht vorgesehen. Für eine vorliegende Clusterung wird der Silhouettenkoeffizient als Mittel der Silhouettewerte ($\text{sil}(l_i)$) aller geclusterten Beobachtungen berechnet, wobei $\text{sil}(l_i)$ über

$$\text{sil}(l_i) = \frac{b(l_i) - a(l_i)}{\max\{a(l_i), b(l_i)\}}$$

definiert ist. Für jede zugeordnete Beobachtung l_i ist dabei $a(l_i)$ die mittlere Unähnlichkeit zwischen l_i und allen übrigen Datenpunkten innerhalb des zugehörigen Clusters, und $b(l_i)$ ist die kleinste mittlere Unähnlichkeit zwischen l_i und den Datenpunkten in den übrigen Clustern. Als Maß für die Unähnlichkeit wird in dieser Arbeit der Euklidische Abstand gewählt. Der Silhouettenkoeffizient kann Werte zwischen -1 und 1 annehmen. Dabei bedeuten negative Werte, dass die entsprechende Beobachtung sich besser in ein anderes Cluster

einfügen würde und bei der Clusterung mit hoher Wahrscheinlichkeit falsch zugeordnet wird. Hohe positive Werte zeigen ein richtiges Clusterungsergebnis an.

3.4. Algorithmus zum Clustern von Graphen

Trotz der zahlreichen Anwendungsmöglichkeiten von Clusterverfahren für Graphen gibt es in diesem Bereich gegenwärtig nur wenige Lösungsvorschläge. Die publizierten Beiträge sind stets auf bestimmte Spezialfälle zugeschnitten wie z.B. die Arbeiten von Gill et al. (2010) oder Altay et al. (2011). Diese sind für Microarrays und Zeitreihen konzipiert und sind in der in dieser Arbeit vorliegenden Situation nicht anwendbar. Eine vielversprechende Herangehensweise schlagen Lohr et al. (2010) vor. In ihrer Arbeit beschreiben und vergleichen sie mehrere Kennzahlen, welche zur Quantifizierung der Unterschiede zwischen Netzwerken herangezogen werden können. Ihr Konzept setzt jedoch voraus, dass die Netzwerkschätzung basierend auf partiellen Korrelationen (vergleiche Abschnitt 2.2) erfolgt. Aus diesem Grund kann diese Arbeit nicht von ihren Ergebnissen profitieren.

In diesem Abschnitt wird ein neues Verfahren zur Berechnung eines Abstands zwischen zwei gerichteten azyklischen Graphen (DAGs) vorgestellt, welches als einzige Einschränkung statische Datenstrukturen voraussetzt. Es kann zur Erstellung einer Abstandsmatrix verwendet werden, die anschließend von allen gängigen Clusteralgorithmen ausgewertet werden kann.

Die folgenden drei Punkte geben mögliche Einsatzbereiche für das im nächsten Abschnitt vorgestellte Verfahren an. In der vorliegenden Arbeit wird es hauptsächlich im Sinne des ersten Punktes genutzt, als eine alternative Möglichkeit, auf die in der generierten MCMC-Stichprobe enthaltenen Informationen zuzugreifen.

Anstatt wie in Abschnitt 3.3 mittels der *pep*-Matrix ein Metamodell aus den generierten DAGs zu erstellen, können diese auch zu Clustern zusammengefasst werden. Die Repräsentanten solcher Cluster sind für den Anwender meist zugänglicher und leichter zu deuten. Ferner ist es so möglich die Ähnlichkeit der in Abschnitt 6.2 betrachteten Netzwerke eingehender zu beurteilen.

Das hier entwickelte Konzept bietet auch die Möglichkeit, neu gefundene Netzwerke mit bereits bekannten Netzwerken automatisch zu vergleichen. Auf diese Weise ist es möglich, die Ergebnisse neuer Experimente direkt mit dem aktuellen Wissensstand in Verbindung zu bringen. Ein automatisierter Abgleich wird immer wichtiger, da es bereits jetzt schon zahlreiche Datenbanken, wie z.B. EcoCyc (Karp et al., 1999), GeneNet (Kolpakov et al., 1999), KEGG (Kanehisa und Goto, 2000) oder RegulonDB (Salgado et al., 2000) gibt, welche hunderte Netzwerkmodelle umfassen und stetig weiter wachsen.

Eine dritte Anwendung, die aber nur angedeutet werden soll, ist im Bereich der medizinischen Diagnose denkbar. Ein aus entnommenem Gewebe inferiertes Netzwerk wird mit Ergebnissen von Gesunden und Kranken verglichen und so einer Gruppe zugeordnet. Dies könnte es ermöglichen auch Netzwerke zu berücksichtigen, deren interne Mechanismen unter Umständen nicht bis ins Letzte erforscht und nachvollzogen worden sind.

Im Folgenden soll die neu vorgeschlagene Abstandsfunktion $dist$,

$$dist : DAG \times DAG \longrightarrow \mathbb{R}^+ \quad ,$$

formal eingeführt werden. Gegeben seien zwei DAGs (DAG_c mit $c = 1, 2$), zwischen denen der Abstand $dist(DAG_1, DAG_2)$ berechnet werden soll. Es gilt $DAG_1 \neq DAG_2$ und $|DAG_1| = |DAG_2|$, wobei $|DAG_c| = d$ die Anzahl der Knoten des DAG_c beschreibt. Für den Fall $DAG_1 = DAG_2$ ist per Definition $dist(DAG_1, DAG_2) = 0$. Ferner gilt, dass DAG_1 aus Datensatz Data_1 und DAG_2 aus Datensatz Data_2 geschätzt wurde. Der Fall $\text{Data}_1 = \text{Data}_2$ ist zulässig. Dieses Szenario zweier unterschiedlicher Netzwerke, die aus einem Datensatz geschätzt worden sind, tritt vor allem dann auf, wenn Netzwerkinferenzverfahren verglichen werden sollen, zum Beispiel NPBN-DP und NPBN-PY, wie es im Abschnitt 6.3 der Fall ist. Beide DAGs liegen als Adjazenzmatrix ($\mathcal{A}^{(c)}$) vor, das heißt als eine quadratische Matrix der Größe d mit den Einträgen 0 und 1, welche die Verbindungen im Graphen kodiert (siehe Abschnitt 2.1, Seite 9). Die Berechnung des Abstands erfolgt iterativ. Zuerst werden alle Nachbargraphen, also Graphen, die sich mit einer einzelnen Kantenoperation konstruieren lassen, aus dem vorliegenden Graphen (o.B.d.A. DAG_1) erzeugt (vergleiche Abschnitt 2.1, Seite 8). Diese werden in einer Menge M_{DAG} gesammelt. Anschließend wird komponentenweise die Differenzmatrix zwischen allen Graphen in M_{DAG} und DAG_2 ge-

bildet. Daraufhin wird die Betragsfunktion auf die erhaltene Differenzmatrix angewendet und die Komponenten werden aufaddiert,

$$\sum_{j}^{d}\sum_{j'}^{d}|\mathcal{A}_{jj'}^{(*)}-\mathcal{A}_{jj'}^{(2)}| \quad \text{mit} \quad DAG_* \in M_{DAG} \ .$$

Der erhaltene Wert beschreibt die Ähnlichkeit zwischen den Elementen aus M_{DAG} und DAG_2. Je niedriger der Wert, desto weniger Unterschiede liegen zwischen den zwei Graphen vor. Anschließend werden alle Elemente aus M_{DAG}, die nicht den niedrigsten Differenzwert zu DAG_2 aufweisen, aus M_{DAG} entfernt. Von den verbliebenen Elementen wird die Likelihood gemäß Gleichung 6 von Seite 18 berechnet. Für ein $DAG_* \in M_{DAG}$ und den Fall Data_1=Data_2 ist $\mathcal{L}(DAG_*|\text{Data}_c)$ wie folgt definiert:

$$\mathcal{L}(DAG_*|\text{Data}_c)=\prod_{j=1}^{d}\int \mathcal{L}\Big(\sigma_j^2,\boldsymbol{\beta}_j|\text{Data}_c^{(\{j\}\cup\mathbf{K}_j)}\Big)p(\sigma_j^2,\boldsymbol{\beta}_j)d\sigma_j d\boldsymbol{\beta}_j \ .$$

Dabei steht d für die Anzahl der Knoten, σ und $\boldsymbol{\beta}$ sind die Parameter der zugehörigen Normalverteilung und $\mathbf{K}_j=\{j^*|X_{j^*}\in \text{pa}_{\mathcal{G}}(X_j)\}$, vergleiche Abschnitt 2.6. Für den Fall, dass $\text{Data}_1 \neq \text{Data}_2$ hat die Likelihood folgende Form:

$$0.5\cdot\Big(\mathcal{L}(DAG_*|\text{Data}_1)+\mathcal{L}(DAG_*|\text{Data}_2)\Big) \ .$$

Dies ist notwendig, um die Symmetrie der *dist* Funktion zu gewährleisten. Nun werden alle DAGs aus M_{DAG} entfernt, die nicht den höchsten Likelihood-Wert aufweisen. Der Wert wird der bislang leeren Menge W hinzugefügt und M_{DAG} wird in M_{DAG}^{alt} umbenannt. Es ist möglich, dass nach diesem Schritt M_{DAG}^{alt} mehr als ein Element enthält, da die Likelihood von äquivalenten Graphen (Graphen, die dieselbe Unabhängigkeitsstruktur repräsentieren) gleich ist (Grzegorczyk und Husmeier (2008)). Damit ist der erste Schritt der Berechnung von $dist(DAG_1, DAG_2)$ abgeschlossen.

Der folgende Schritt wird gegebenenfalls mehrfach ausgeführt. Es wird eine neue Menge M_{DAG} gebildet, welche alle Nachbargraphen der DAGs aus M_{DAG}^{alt} enthält. Aus dieser Menge werden alle DAGs entfernt, die in einer vorigen Iteration bereits betrachtet wurden. Dies soll Endlosschleifen verhindern und die Konvergenz des Algorithmus garantieren. Anschließend wird, nach Entfernen der unähnlichen Elemente, wie zuvor auf Grundlage der komponentenweisen Differenz zu DAG_2, die Likelihood berechnet. Dann werden aus

M_{DAG} alle DAGs entfernt, die nicht die höchste Likelihood haben, der Wert der Likelihood der Menge W hinzugefügt und M_{DAG} wird in M_{DAG}^{alt} umbenannt. Dieser Vorgang wird solange wiederholt, bis die Menge M_{DAG} einen DAG enthält, für den die Differenz zu DAG_2 für sämtliche Komponenten gleich null ist. Tritt dies ein, wird die Berechnung beendet. Die Summe der in Menge W enthaltenen Werte bildet das Ergebnis der Berechnung von $dist(DAG_1, DAG_2)$.

Anschließend wird die für die Clusterung notwendige, sogenannte Abstandsmatrix erstellt. Seien DAG_c, $c \in \{1, \ldots, C\} \subset \mathbb{N}$, die zu clusternden DAGs. Zuerst wird von allen möglichen Paaren der Abstand bestimmt $dist(\mathrm{DAG}_c, \mathrm{DAG}_{c^*}), \forall c, c^* \in \{1, \ldots, C\}$. Die zu bildende C-dimensionale Abstandsmatrix ist quadratisch und symmetrisch. In der c-ten Spalte und der $c*$-ten Zeile steht der Abstand zwischen DAG_c und DAG_{c^*}. Für das eigentliche Clustern kann jedes gängige Verfahren verwendet werden, welches mit Abstandsmatrizen umgehen kann. In dieser Arbeit wird die hierarchische Clusteranalyse nach Ward (Ward, 1963) verwendet. Diese ist in R implementiert und über den Befehl `hclust` abrufbar. Vergleiche auch die Beschreibung der hierarchische Clusteranalyse im Abschnitt 4.2.

Eine umfassende Beschreibung der algorithmischen Umsetzung der hier verwendeten und im Rahmen dieser Arbeit entstandenen Abstandsfunktion befindet sich im Anhang im Abschnitt A ab Seite 96.

Die Grundidee des vorgestellten Konzeptes die Anzahl der benötigten Operationen, um ein Objekt in ein anderes zu überführen, als Maß der Ähnlichkeit zu verwenden, ist für sich genommen nicht neu. Dieses Konzept ist in unterschiedlichen, auch gewichteten, Variationen unter der Sammelbezeichnung Editierdistanz geläufig. Der in dieser Arbeit vorgestellte Ansatz arbeitet ohne Gewichtung der verschiedenen Kantenoperationen. Damit ist er am besten mit der „Levenshtein distance“ (Levenshtein, 1966) vergleichbar. In zwei Punkten unterscheidet er sich aber von dieser wesentlich. Es sind nur Kantenoperationen erlaubt, welche zu gemäß der Definition eines DAGs gültigen Graphen führen und in jeder Iteration wird die Likelihood der gefundenen Graphen überprüft. Somit werden bei der Berechnung keine in Sinne der Ausgangsdaten unplausiblen Graphen berücksichtigt.

4. Analyse und Vergleich der Leistungsfähigkeit der Verfahren am Erk-Signalübertragungsnetzwerk-Modell

In diesem Abschnitt wird im Rahmen einer Simulationsstudie die Leistungsfähigkeit des vorgeschlagenen Verfahrens überprüft und mit den Referenzverfahren k-means und hierarchische Clusteranalyse verglichen. Dabei wird sowohl auf die Qualität der Entmischung als auch auf die Genauigkeit bei der Schätzung der Clusteranzahl eingegangen. Zunächst werden die Simulation und die Datenstruktur besprochen, danach werden die Bewertungskriterien festgelegt und anschließend das Verfahren für beide a priori Prozesse (Dirichlet-Prozess, vergleiche Abschnitt 3.1.2 und Pitman-Yor-Prozess, vergleiche Abschnitt 3.1.3) bewertet.

4.1. Datengenerierung

Zwecks einer umfassenden Beurteilung der Methoden werden mittels Simulation realitätsnahe Datensätze erstellt. Als Grundlage wird das etablierte, von Sasagawa et al. (2005) vorgeschlagene dynamische Erk-Signalübertragungsnetzwerk-Modell für PC12 Zellen gewählt. Dieses ist im Systems Biology Markup Language Format auf den Seiten des Europäischen Bioinformatik-Instituts (www.ebi.ac.uk, BIOMD0000000049, Stand 24.10.2011) abrufbar. Das Modell umfasst 99 Spezies, 150 Reaktionen sowie 234 Reaktionsparameter, welche die Reaktionen spezifizieren. Es bildet nach, wie ein externer Reiz durch die Zelle verarbeitet wird. Dies äußert sich durch deutliche Änderung der Molekülkonzentrationen im Zellinneren und geht mit der Ausprägung einer bestimmten Interaktionstopologie eines Netzwerks einher (Wieczorek et al., 2015). Eine vereinfachte Darstellung eines Ausschnittes des Netzwerks ist in Abbildung 3 auf Seite 41 zu finden. Die Konzentration der Botenstoffe Epidermaler Wachstumsfaktor, im Folgendem EGF, und Nerven-Wachstumsfaktor, im Folgendem NGF, dient hierbei als Reizübermittler und veranlasst die simulierte Zelle beim Überschreiten bestimmter Konzentrationsschwellen zu differenzierten Reaktionen (Santos et al., 2007).

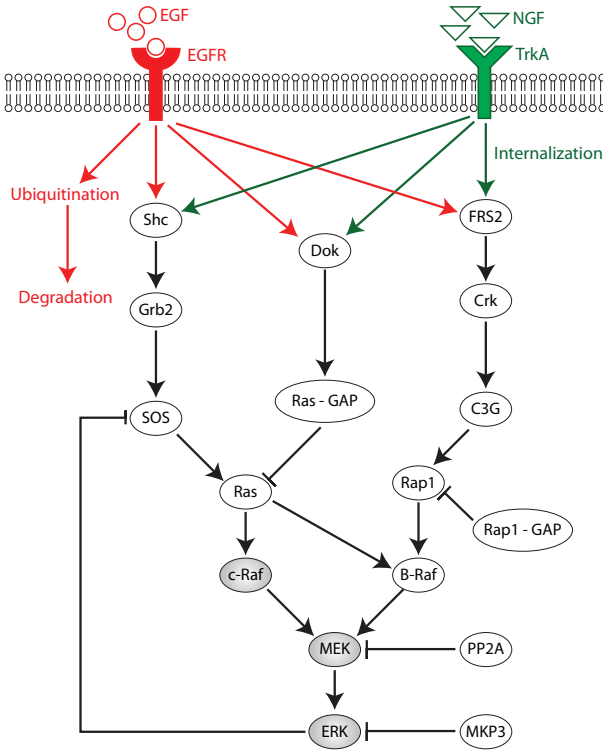


Abbildung 3: *Schematische Darstellung des simulierten Erk-Signalübertragungsnetzwerkes. Abgebildet ist ein Ausschnitt aus dem Netzwerk, welcher stark vereinfacht die Reaktion der Zelle auf die chemischen Botenstoffe EGF (rot hervorgehoben) und NGF (grün hervorgehoben) darstellt. Die für diese Arbeit zentralen Spezies Raf (c-Raf), MEK und ERK sind grau unterlegt. Quelle: Wieczorek et al. (2015).*

Das von Sasagawa et al. (2005) beschriebene biochemische Modell wird erweitert. Es umfasst nun zwei Typen von Zellen: den in der Natur vorkommenden Wildtyp und eine am MPI Dortmund simulierte Mutante mit geringerer katalytischen Aktivität von MEK. Im Modell wird diese durch den Reaktionsparameter J136 gesteuert. Während dieser Parameter beim Wildtyp 0.15 μM/s beträgt, ist die Einstellung für mutierte Zellen gleich 0.015 μM/s. Durch die Vorgabe des jeweiligen Zelltyps und einer bestimmten Konzentration von EGF und NGF können somit Beobachtungen aus vier unterschiedlichen Interaktionstopologien erzeugt werden: Wildtyp mit EGF stimuliert, Wildtyp mit NGF stimuliert, Mutante mit EGF stimuliert und Mutante mit NGF stimuliert (Wieczorek et al., 2015).

Eine schematische Darstellung der simulierten Netzwerke mit einer Gegenüberstellung von Wildtyp und Mutante bietet Abbildung 16 im Anhang auf Seite 115. Die Simulation bildet die biologischen Vorgänge in sehr detaillierter Weise nach. Die Konzentrationen sämtlicher beteiligter Moleküle wird berechnet und steht zur Verfügung. In dieser Arbeit wird jedoch nicht das gesamte Netzwerk modelliert. Durch Expertenwissen können drei zentrale Spezies identifiziert werden, welche die unterschiedlichen Interaktionstopologien wiedergeben. Dabei handelt es sich um die phosphorylierte rapidly-growing-fibrosarcoma kinase (Bezeichnung nach Sasagawa: c_Raf_Ras_GTP, im Folgendem RAF), die doppelt phosphorylierte extracellular-signal-regulated kinase (ppERK, im Folgendem ERK) und um die mitogen-activated protein kinase (ppMEK, im Folgendem MEK). Die Festlegung auf die Betrachtung von drei Spezies/Knoten hat mehrere Gründe. Drei Spezies sind die Anzahl, welche sich beim gegenwärtigen Stand der Technik in ausreichender Stichprobengröße erheben lässt. Des Weiteren steht in dieser Arbeit die statistische Methode im Vordergrund. Drei Knoten sind das einfachste denkbare Netzwerk, an dem das NPBN-Verfahren sinnvoll erprobt werden kann. Dabei ist neben der Netzwerkmodellierung die Entmischung der in den Daten verborgenen Subgruppen vom großem Interesse. Diese wird, anders als die Netzwerkmodellierung, durch eine geringe Anzahl an Knoten sogar tendenziell erschwert, da der Algorithmus weniger Informationen zur Verfügung hat. Hinzu kommt, dass ein Netzwerk mit 99 Knoten in seinem Umfang für die in dieser Arbeit vorgeschlagene NPBN-Analyse zu groß und nur schwer handhabbar wäre.

Der NPBN-Algorithmus kann auf handelsüblichen Rechnern bei vertretbarer Rechenzeit Netzwerke von bis zu 50 Knoten analysieren. Der Einsatz großer Rechnercluster bietet jedoch die Möglichkeit die Knotenanzahl deutlich zu erhöhen.

Der generierte Datensatz umfasst also stellvertretend für alle die Konzentrationen von drei für die Signalverarbeitungskette zentralen Spezies: RAF, MEK und ERK. Diese werden von der ersten bis zur zehnten Minute nach der Stimulation mit EGF und NGF alle 30 Sekunden erhoben. Da sich aufeinander folgende Beobachtungen nur geringfügig voneinander unterscheiden und um den Rechenaufwand zu senken, wird die Analyse auf die Messungen der vollen Minuten beschränkt. Nach zehn Minuten gilt aus biochemischer Sicht die

Signalweiterleitung als abgeschlossen und die Proteinkonzentrationen in der Zelle befinden sich nahe dem Ausgangsniveau von Minute eins, vergleiche Abbildung 4 a)-c) auf Seite 45.

Für jede Beobachtung wird die Simulation neu initiiert, indem die Ausgangswerte der Konzentration von RAF, MEK und ERK aus einer Normalverteilung gezogen werden

$$\begin{aligned} \mathrm{RAF}_{\mathrm{Ausgangskonzentration}} &\sim \mathrm{N}(0.5, 0.5 \cdot \eta) \\ \mathrm{MEK}_{\mathrm{Ausgangskonzentration}} &\sim \mathrm{N}(0.68, 0.68 \cdot \eta) \\ \mathrm{ERK}_{\mathrm{Ausgangskonzentration}} &\sim \mathrm{N}(0.26, 0.26 \cdot \eta) \ . \end{aligned}$$

Die Erwartungswerte wurden dabei aus dem Modell von Sasagawa et al. (2005) übernommen. Unplausible negative Werte wurden durch Anwendung der Betragsfunktion korrigiert. Der Parameter $\eta \in (0.1, 0.2, \ldots, 0.7)$ wird eingeführt, um biologisches Rauschen in der Simulation zu berücksichtigen. Diese Größe wird im Folgenden Noise genannt. Höhere Einstellungen von η ($\eta > 0.7$) sind nach Expertenwissen äußerst unplausibel, da eine Zelle, deren innerer biochemischer Zustand derart vom Normalniveau abweicht, nicht überlebensfähig ist.

Durch Variation des Parameters η können Zellen im „Ruhezustand“ simuliert werden, mit klaren und für die Inferenz günstigen Profilen, aber ebenso, bei höheren η, „aktive“ Zellen, deren Profile uneinheitlich sind und bei denen viele Inferenzmethoden an ihre Grenzen stoßen und falsche Ergebnisse liefern. Die Abhängigkeit der Varianz vom Erwartungswert sorgt dafür, dass sich die Schwankung in den Werten proportional zu ihrer mittleren Lage verhält. Die vollständige Aufzählung aller im Modell vorkommenden Spezies sowie deren Ausgangskonzentrationen sind im Anhang in Tabelle 5 ab Seite 113 zu finden. Die vollständige Liste aller im Modell betrachteten Reaktionen kann Tabelle 6 ab Seite 123 entnommen werden. Weitere Details der Simulation, darunter das von Sasagawa als Ausgangspunkt verwendete Differentialgleichungssystem, werden im Anhang im Abschnitt D beschrieben. Es werden zwei, zehn Zeitpunkte umfassende, Datensätze simuliert, D2 und D4. D2 enthält Beobachtungen eines Netzwerks mit zwei Komponenten (Wildtyp stimuliert mit NGF oder EGF). D4 enthält Beobachtungen eines Netzwerks mit vier Komponenten (Wildtyp und Mutante jeweils mit NGF oder EGF stimuliert). Der erste Datensatz umfasst pro Zeitpunkt 350 Beobachtungen, der zweite 700. Diese werden jeweils

gleichmäßig auf die Komponenten aufgeteilt. Weitere Erläuterungen und Beispiele zu der Struktur der Daten können Tabelle 1 sowie den Abbildungen 4 auf Seite 45 und 16 auf Seite 115 entnommen werden.

Anzahl simulierter Beobachtungen für D2

Typ	Stimulation	simulierte Noisestärke						
	mit	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Wildtyp	NGF	175	175	175	175	175	175	175
	EGF	175	175	175	175	175	175	175
	$\sum$	350	350	350	350	350	350	350

Anzahl simulierter Beobachtungen für D4

Typ	Stimulation	simulierte Noisestärke						
	mit	0.1	0.2	0.3	0.4	0.5	0.6	0.7
Wildtyp	NGF	175	175	175	175	175	175	175
	EGF	175	175	175	175	175	175	175
Mutante	NGF	175	175	175	175	175	175	175
	EGF	175	175	175	175	175	175	175
	$\sum$	700	700	700	700	700	700	700

Tabelle 1: *Übersicht der simulierten Beobachtungen für das Netzwerk mit zwei Komponenten (D2) und das Netzwerk mit vier Komponenten (D4). Die dargestellte Konstellation trifft auf alle der 10 simulierten Zeitpunkte in D2 und D4 zu.*

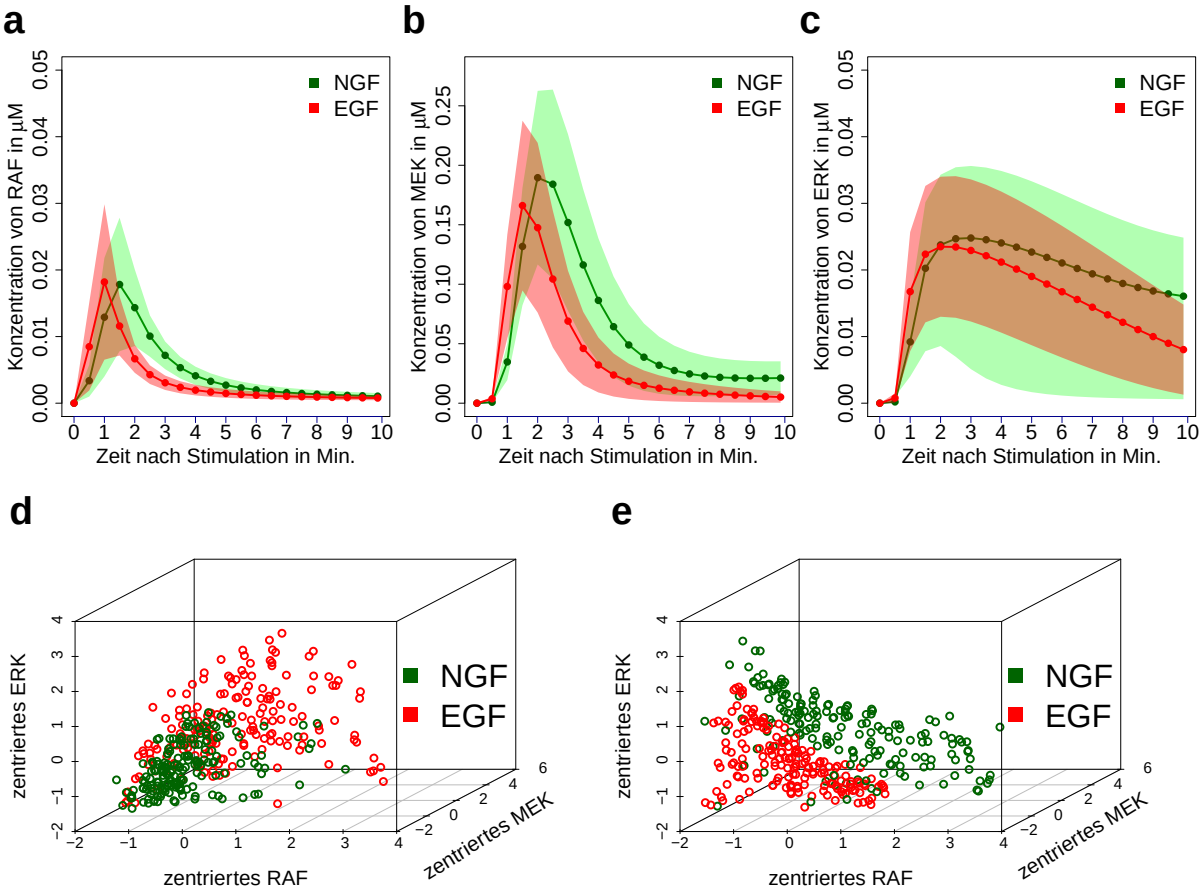


Abbildung 4: *Struktur der simulierten Daten bei Noise 0.7, bei Stimulation mit NGF (grün) und mit EGF (rot), im zwei Komponenten Fall. Simulierte Konzentrationen der Proteine RAF (a), MEK (b), ERK (c) gegen die Zeit aufgetragen; die feste Linie kennzeichnet die Mittel der Werte, der schraffierte Bereich die Standardabweichung. Simulierte Konzentrationen von RAF, MEK und ERK (die Werte sind in der Abbildung zentriert dargestellt), 2 Min. nach Stimulation (d), 9 Min. nach Stimulation (e).*

4.2. Referenzverfahren

Mittlerweile wurden alternative Vorschläge für Verfahren publiziert, welche selbstständig die Zahl der Gruppen im Datensatz schätzen und die Beobachtungen diesen zuordnen können, wie zum Beispiel das von Hasenauer et al. (2014), welches Mischungsmodelle mit gewöhnlichen Differentialgleichungen kombiniert. Zu dem Zeitpunkt, als der in dieser Arbeit vorgestellte NPBN-Ansatz ausgearbeitet worden ist, war in der Literatur kein alternatives Verfahren mit diesen Eigenschaften bekannt. Aus diesem Grund wird die Leistungsfähigkeit von NPBN mit der von Verfahren aus der Clusteranalyse verglichen.

Eine korrekte Clusterung liegt dann vor, wenn Beobachtungen, die aus einen Netzwerk stammen, in das gleiche Cluster eingeordnet werden. Um aussagekräftige Resultate zu gewährleisten, wird den Referenzverfahren ein Informationsvorsprung gewährt, indem sie mit der korrekten Anzahl von Gruppen initiiert werden, während das NPBN-Verfahren diese selbstständig ermittelt.

Das erste Referenzverfahren ist der k-means Algorithmus (MacQueen, 1967), im Folgenden KM, welcher einen Datensatz in k (k durch den Benutzer vorgegeben) Cluster partitioniert, so dass die Summe der Abstände aller Beobachtungen zu den zugehörigen arithmetischen Clustermitteln minimiert wird. Die in dieser Arbeit diskutierten Ergebnisse werden in Matlab (MATLAB, 2009) unter Verwendung der Funktion `kmeans.m` mit dem Euklidischen Abstandsmaß berechnet. Jede Clusterung wird 500 mal mit zufälligen Startwerten wiederholt, um so die zufällige Initiierung des Verfahrens aufzuwiegen.

Das zweite Referenzverfahren, das herangezogen wird, ist die hierarchische Clusteranalyse, im Folgenden HC. Dabei handelt es sich um ein agglomeratives Verfahren, welches erst Paare bzw. kleine Mengen von Beobachtungen bildet und diese durch Hinzunahme weiterer Beobachtungen oder Vereinigung der Mengen untereinander schrittweise vergrößert, bis der gesamte Datensatz in einem einzigen Cluster vereint ist. In jedem Schritt werden diejenigen Beobachtungen zusammengeführt, welche minimalen Abstand zueinander haben. Auf jeder Stufe lässt sich eine Clusterung mit vorgegebener Clusteranzahl ableiten. Die hierarchische Clusteranalyse wird ebenfalls in Matlab unter Verwendung der Funktionen `pdist.m` und

`linkage.m` mit dem Euklidischen Abstandsmaß nach der Methode von Ward (Ward, 1963) durchgeführt.

4.3. Vergleich der Clustergüte anhand der Reinheit

An dieser Stelle wird die Leistungsfähigkeit der Entmischung von Beobachtungen, die aus verschiedenen Komponenten stammen, betrachtet. Verglichen werden NPBN-DP und NPBN-PY sowohl untereinander als auch mit den Referenzmethoden. Die vorgestellten Ergebnisse stammen aus MCMC-Simulationen mit $2.8 \cdot 10^6$ Iterationen mit einer Ausdünnung (thinning) von 350 und einem Burn-In von $1.4 \cdot 10^6$ Iterationen für Netzwerke mit zwei Komponenten und aus Simulationen mit $5 \cdot 10^6$ Iterationen mit einer Ausdünnung von 500 und einem Burn-In von $2 \cdot 10^6$ Iterationen für Netzwerke mit vier Subgruppen. Diese Einstellungen werden für alle zehn betrachteten Zeitpunkte für beide Varianten von NPBN verwendet. Als Bewertungskriterium dient die in Abschnitt 3.3 beschriebene Kennzahl (*pco*). Für eine bessere Übersicht werden die berechneten *pco*-Werte der vier in der Simulationsstudie zu vergleichenden Verfahren nach η gruppiert und in Boxplots zusammengefasst. Jeder Boxplot beinhaltet somit die Entmischungsergebnisse für alle zehn Zeitpunkte der simulierten Signalverarbeitung.

Abbildung 5 auf Seite 48 zeigt die *pco*-Werte für den D2 Datensatz. Es ist erkennbar, dass bei geringer Noisestärke alle Verfahren gleich gut sind und hohe *pco*-Werte nahe bei 100 % liefern. Bei einer Noisestärke von 0.3 beginnt sich ein Unterschied in der Leistungsfähigkeit der Methoden abzuzeichnen. Während die Ergebnisse der klassischen Clusterverfahren (KM und HC), welche die Netzwerkstruktur nicht berücksichtigen können, stark schwanken, bleiben diese für beide NPBN-Verfahren bei über 95 %. Bei höheren Noisestärken fällt für die Referenzverfahren der Median der *pco*-Werte bis auf unter 65 %, während die Streuung steigt. Die NPBN-Verfahren sind von der steigenden Noisestärke auch negativ betroffen. Ihre Leistung wird ebenfalls schlechter, jedoch in einem deutlich kleineren Umfang. Selbst bei moderatem Noise von 0.4 liegen deren *pco*-Werte deutlich über 90 %, mit geringer Streuung, und auch bei noch höherem Noiseniveau ist das Ergebnis (Median und Interquartilsabstand) stets deutlich besser als das der klassischen reinen

Clusteralgorithmen. Die große Streuung der *pco*-Werte, die sich insbesondere bei den klassischen Verfahren bemerkbar macht, ist dadurch zu erklären, dass die Beobachtungen, die für unterschiedliche Zeitpunkte erhoben wurden, von den Verfahren unterschiedlich gut entmischt werden können. Das liegt daran, dass die simulierten Proteinkonzentrationen in der mittleren Phase der Signalverarbeitung dicht beieinander liegen und stark überlappen (vergleiche die Zeitpunkte 5.-7. Min. in Abbildungen 4 (a-c) auf Seite 45) sowie in Abbildungen 9 und 10 ab Seite 56) und die Netzwerke schwerer zu unterscheiden sind, als dies der Fall am Anfang, kurz nach der Stimulation, und am Ende der Signalverarbeitung ist (1. und 2. sowie 9. und 10. Min.).

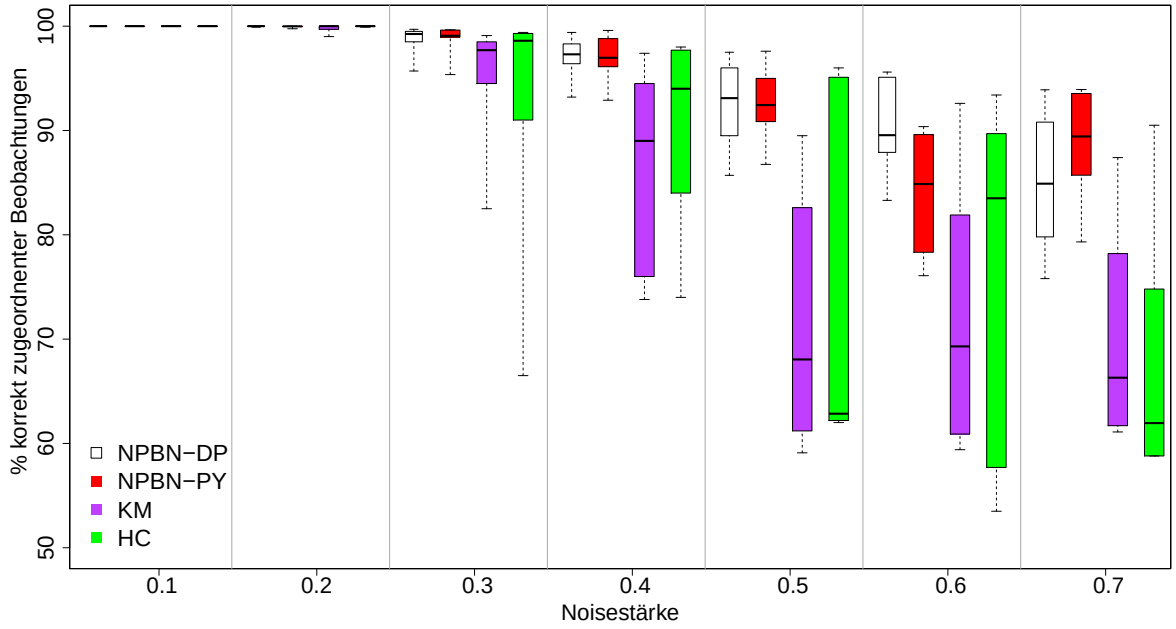


Abbildung 5: *Vergleich der korrekt allokierten Beobachtungen in Prozent für aufsteigende Noisestärke, zusammengefasst über die Zeitpunkte für den Datensatz mit zwei Komponenten (D2). Abgebildet sind die Ergebnisse für die nichtparametrischen Bayesschen Netzwerke mit Dirichlet a priori Verteilung (NPBN-DP,* $\theta = 1$*) und mit Pitman-Yor a priori Verteilung (NPBN-PY,* $\theta = 1,\ \sigma = 0.2$*) (vergleiche Abschnitt 3.2) sowie die Ergebnisse der beiden Referenzverfahren k-means (KM) und hierarchische Clusteranalyse (HC), vergleiche Abschnitt 4.2.*

Kurz nach der Stimulation und am Ende der Signalverarbeitung herrscht nur ein geringes Ausmaß an Aktivität und die Interaktionsstruktur der beiden Netzwerke unterscheidet sich deutlicher voneinander. Dies ist ein weiterer Grund die Ergebnisse über die Zeitpunkte zusammengefasst zu betrachten. Auf diese Weise sollen aussagekräftige, belastbare Vergleiche ermöglicht werden, da sowohl günstige als auch ungünstige Datenkonstellationen in die Bewertung einfließen. Die Abbildungen 4 (d-e) auf Seite 45 verdeutlichen den beschriebenen Sachverhalt anhand von Beispielen.

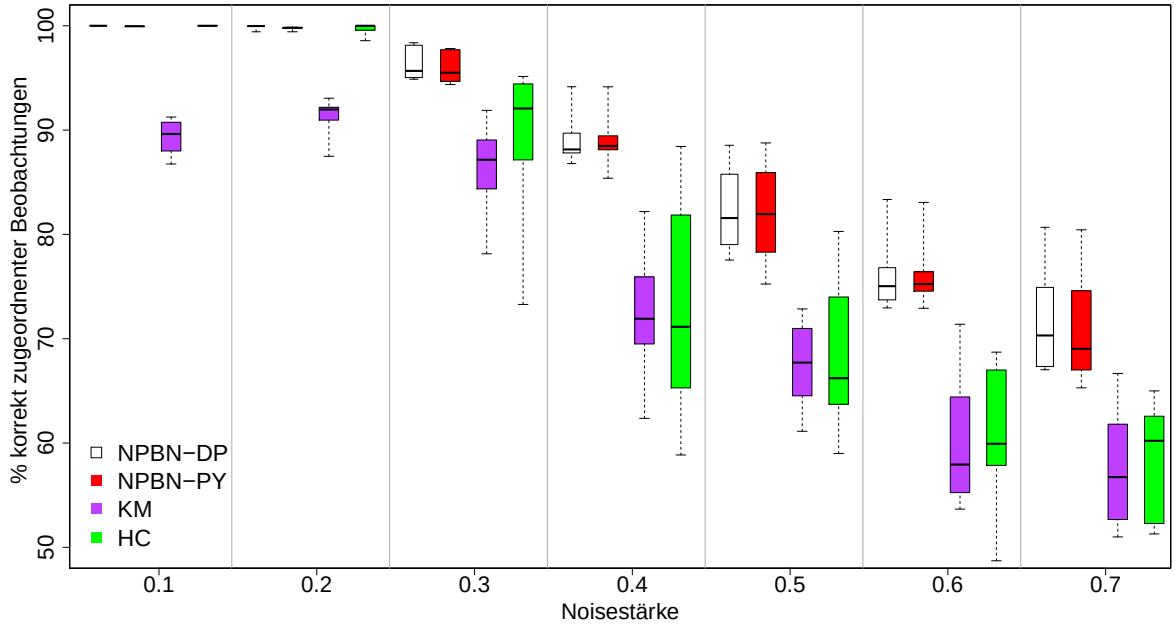


Abbildung 6: *Vergleich der korrekt allokierten Beobachtungen in Prozent für aufsteigende Noisestärke über die Zeitpunkte, zusammengefasst für den Datensatz mit vier Komponenten (D4). Abgebildet sind die Ergebnisse für die nichtparametrischen Bayesschen Netzwerke mit Dirichlet a priori Verteilung (NPBN-DP, $\theta = 1$) und mit Pitman-Yor a priori Verteilung (NPBN-PY, $\theta = 1,\ \sigma = 0.2$) (vergleiche Abschnitt 3.2) sowie die Ergebnisse der beiden Referenzverfahren k-means (KM) und hierarchische Clusteranalyse (HC), vergleiche Abschnitt 4.2.*

Die Analyse des Datensatzes mit vier Komponenten (D4, vergleiche Abbildung 6 auf Seite 49) bestätigt im Wesentlichen die Ergebnisse aus der Analyse des Datensatzes mit zwei

Komponenten (D2). Die NPBN-Verfahren liefern teilweise deutlich bessere Ergebnisse als die Referenzverfahren. Allerdings scheint der Einfluss der Noisestärke im D4 Datensatz ausgeprägter zu sein als dies im D2 Datensatz der Fall war. Besonders auffällig ist dies bei der Betrachtung der Ergebnisse des k-means Verfahrens. Selbst bei gering ausgeprägter Noisestärke erreichen sie Maximalwerte im Bereich von lediglich 90 %. Auch der *pco*-Wert der hierarchischen Clusteranalyse ist schlechter als im D2 Datensatz. Bereits bei $\eta = 0.2$ liegt er unter 100 % und sinkt mit zunehmendem Noise stark ab. Generell fallen die Werte aller Verfahren deutlich schneller als dies bei D2 der Fall war. Dies ist insofern nachvollziehbar, als hier Beobachtungen anhand von drei Werten in vier Gruppen aufgeteilt werden sollen. Ohne die Zusatzinformationen, welche die NPBN-Verfahren aus der Netzwerkstruktur ziehen können, lassen sich die mit steigendem Noise immer stärker überlappenden Gruppen kaum trennen. Auf den höchsten Einstellungen von 0.6 und 0.7 werden die Ergebnisse aller Verfahren schlechter und fallen auf 75 % und weniger. Die von den Clusteralgorithmen gelieferten Werte liegen hier mit 60 % und weniger in einen Bereich, in dem sie kaum noch besser sind als bei einer zufälligen Aufteilung der Beobachtungen.

4.4. Vergleich der gefundenen Komponentenanzahl

Neben der Clustergüte, welche im vorhergehenden Abschnitt betrachtet wird, ist auch die korrekte Schätzung der Anzahl an Komponenten ein wichtiges Kriterium für die Leistungsfähigkeit der in dieser Arbeit vorgestellten NPBN-Verfahren. Diese wird im Folgendem mit Hilfe des im Abschnitt 3.3.3 vorgestellten Silhouettenkoeffizienten untersucht.

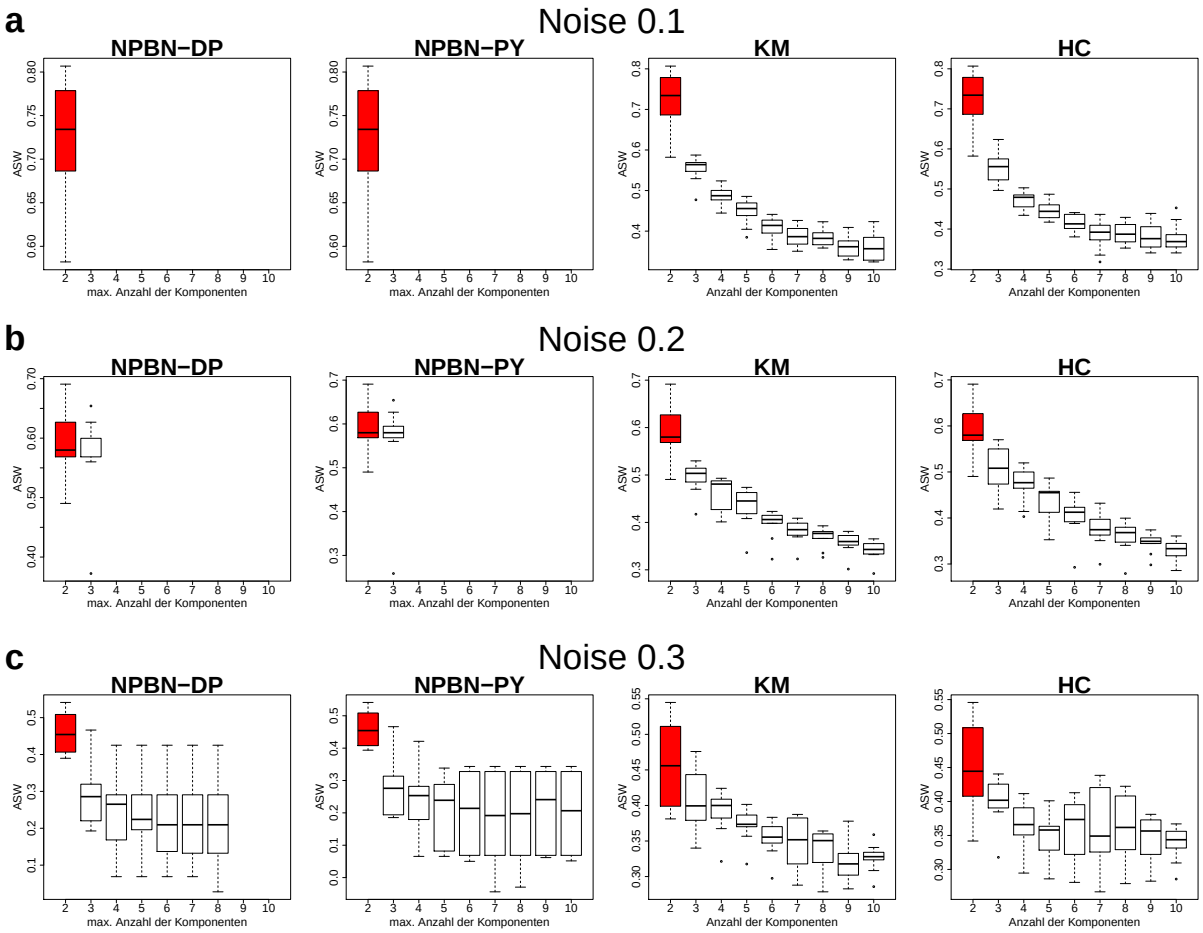
Die Abbildungen 7 und 8 auf den Seiten 53 und 55 zeigen die ASW-Werte, welche für zunehmende Noisestärke von 0.1 bis 0.7 für die Datensätze D2 und D4 berechnet wurden. Zusätzlich wird die Einstellung von NPBN-DP, NPBN-PY, k-means und hierarchischer Clusteranalyse in Bezug auf die für die Aufteilung der Beobachtungen maximal zugelassene Clusteranzahl zwischen zwei und zehn variiert. Die Werte werden, wie zuvor im Abschnitt 4.3, über die Zeitpunkte zusammengefasst. Die korrekte Komponentenanzahl (zwei in Abbildung 7, vier in Abbildung 8) ist rot markiert. Die NPBN-Verfahren schöpfen bei geringen Noisestärken die maximale Komponentenanzahl nicht aus. Das führt dazu,

dass beispielsweise eine Berechnung mit fünf oder sechs maximal erlaubten Komponenten identisch ist mit jener, in der nur vier Komponenten zugelassen sind. In solchen Fällen werden die sich wiederholenden Boxplots nicht eingezeichnet.

Außer im Fall von Noisestärke 0.3, wo die hierarchische Clusteranalyse den höchsten ASW-Wert bei 3 Clustern statt bei 4 aufweist, zeigen alle Verfahren bei geringer Noisestärke von 0.1 bis 0.3 die richtige Komponentenanzahl an, sowohl im D2 wie auch im D4 Datensatz.

Bei höheren Noisestärken bietet sich ein differenzierteres Bild. Im Zwei-Komponenten-Fall unterliegen die klassischen Clusterverfahren hohen Schwankungen und liefern häufig falsche Ergebnisse, wie z.B. im Fall von Noisestärke 0.6 (Abbildung 7(f)). Hier sind die ASW-Werte der hierarchischen Clusteranalyse für den D2 Datensatz selbst für fünf Cluster höher als die ASW-Werte für die korrekten zwei Cluster. Bei den NPBN-Verfahren (NPBN-DP und NPBN-PY) sind die ASW-Werte durchgehend bei der korrekten Zahl von zwei Clustern am höchsten. Es finden sich keine Hinweise darauf, dass eines der beiden NPBN-Verfahren die Clusteranzahl besser als das andere schätzen kann.

Im Vier-Komponenten-Fall liefern die klassischen Methoden bereits bei moderater Noisestärke von 0.4 in den meisten Fällen die falsche Clusteranzahl. Die NPBN-Verfahren sind im Vier-Komponenten-Fall zwar nicht mehr ganz fehlerfrei, liefern aber insgesamt noch gute Ergebnisse und liegen im Median richtig. Hier gibt es auch Anzeichen dafür, dass das NPBN-PY leicht bessere Ergebnisse als das NPBN-DP liefert, wie z.B. im Fall von Noisestärke 0.4 (Abbildung 8(d)), in dem das NPBN-DP mit zwei Komponenten einen falschen Wert liefert.



Diese Abbildung wird auf der folgenden Seite fortgesetzt.

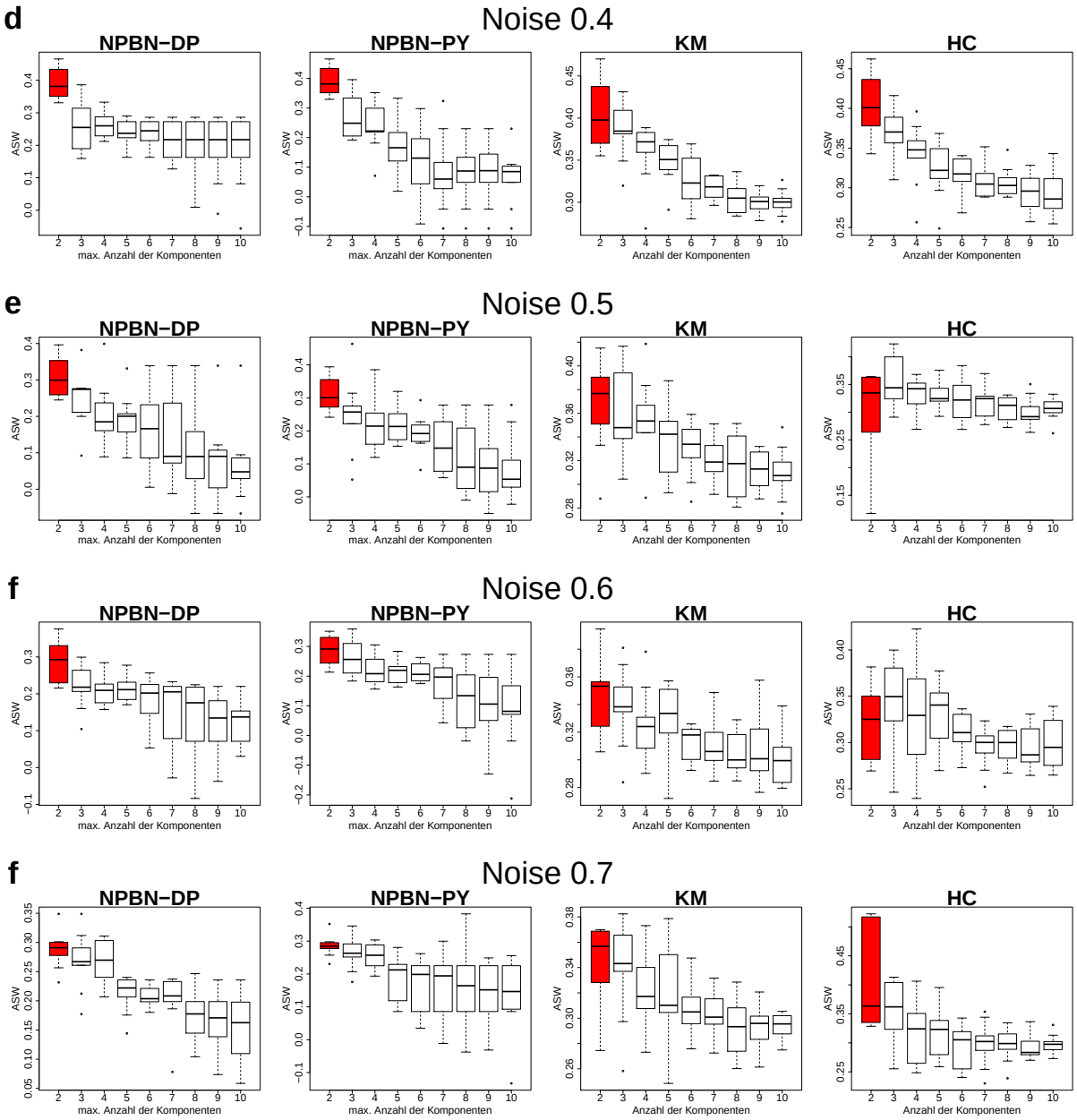
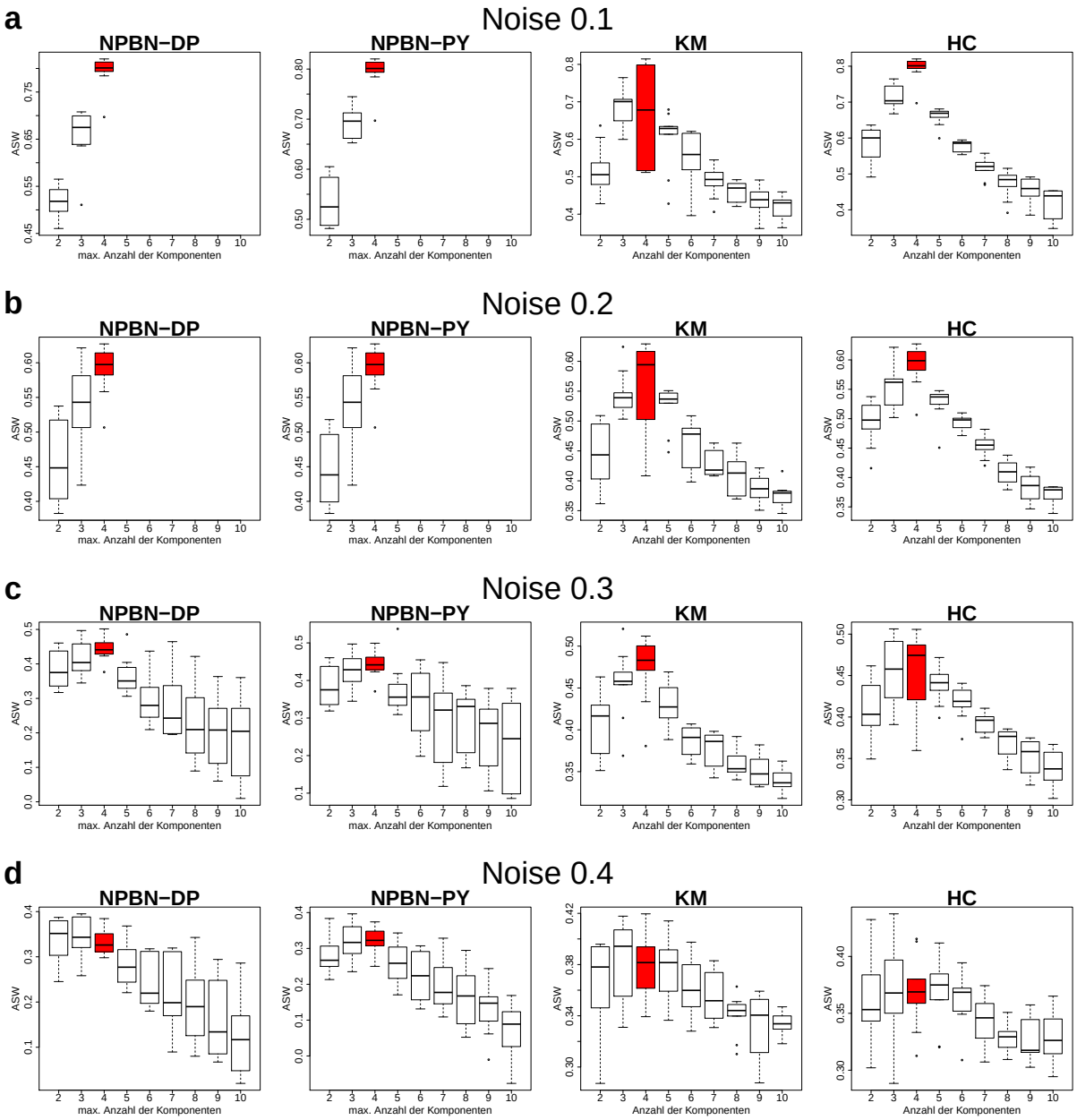


Abbildung 7: *Über die Zeitpunkte zusammengefasste Silhouettenkoeffizienten, berechnet für aufbereitete Allokationsvektoren bei vorgegebener maximaler Komponentenzahl bzw. Clusteranzahl, für den simulierten Datensatz mit zwei Komponenten (D2). Verglichen werden NPBN-DP, NPBN-PY, k-means und hierarchisches Clustern bei variierender Noisestärke. Noise 0.1 (a), Noise 0.2 (b), Noise 0.3 (c), Noise 0.4 (d), Noise 0.5 (e), Noise 0.6 (f), Noise 0.7 (g). Die Box, die für die korrekte Anzahl der Cluster (hier zwei) steht, ist rot hervorgehoben.*



Diese Abbildung wird auf der folgenden Seite fortgesetzt.

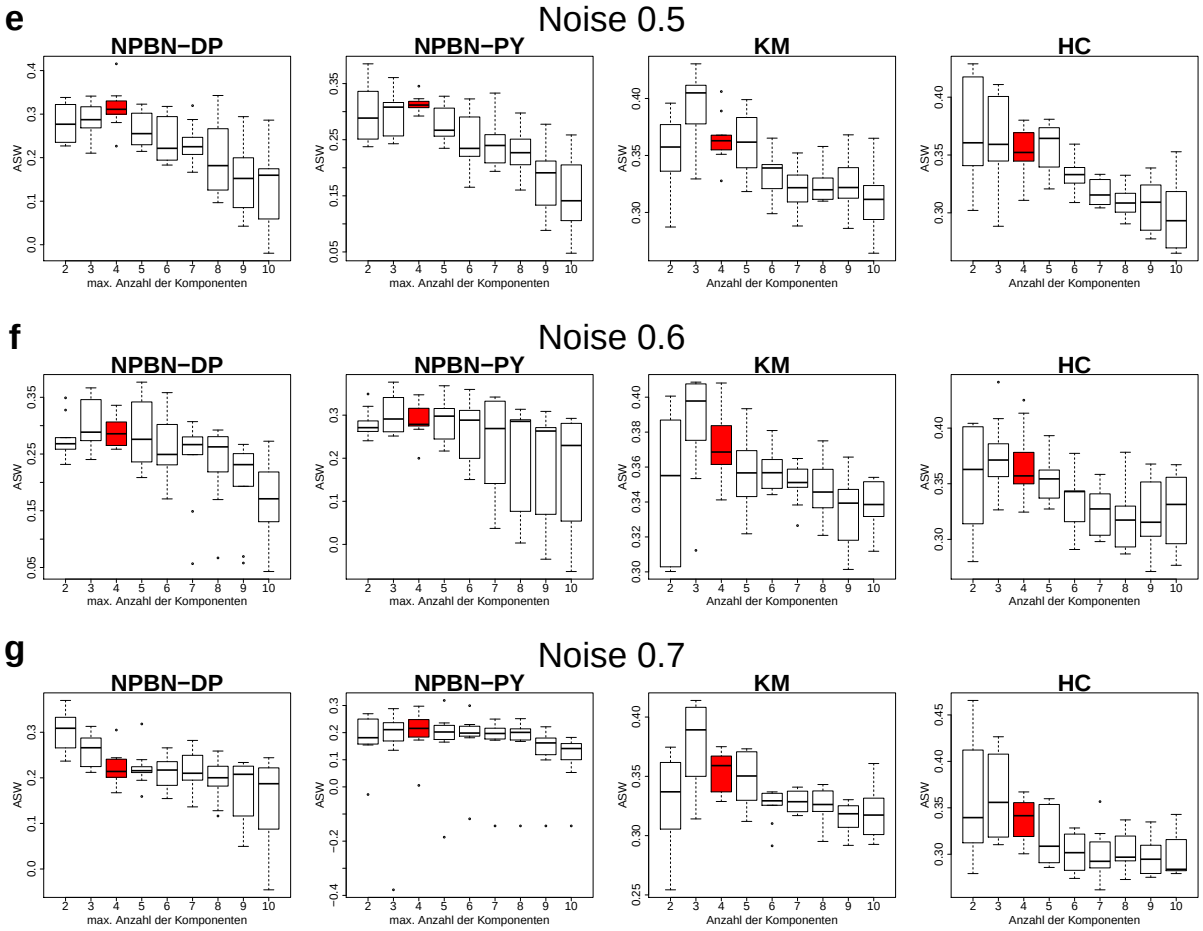


Abbildung 8: *Über die Zeitpunkte zusammengefasste Silhouettenkoeffizienten berechnet für aufbereitete Allokationsvektoren bei vorgegebener maximaler Komponentenzahl bzw. Clusteranzahl, berechnet für den simulierten Datensatz mit vier Komponenten (D4). Verglichen werden NPBN-DP, NPBN-PY, k-means und hierarchisches Clustern bei variierender Noisestärke. Noise 0.1 (a), Noise 0.2 (b), Noise 0.3 (c), Noise 0.4 (d), Noise 0.5 (e), Noise 0.6 (f), Noise 0.7 (g). Die Box, die für die korrekte Anzahl der Cluster (hier vier) steht, ist rot hervorgehoben.*

4.5. Vergleich der Leistung von NPBN-DP und NPBN-PY

Für beide Datensätze gilt, dass die Ergebnisse von NPBN-DP und NPBN-PY nahe beieinander liegen. Die einzige Abweichung ist für das Zwei-Komponenten-Netzwerk für Noise 0.6 und 0.7 zu beobachten. Jedoch ist hier jedes Verfahren dem anderen mal über-, mal unterlegen, so dass von einer zufälligen Schwankung ausgegangen werden kann. Eine wiederholte Schätzung mit einer anderen Stichprobe aus dem Datensatz liefert vergleichbare Werte. Hier weichen die von den Schätzern gelieferten Werte nur um wenige Nachkommastellen voneinander ab.

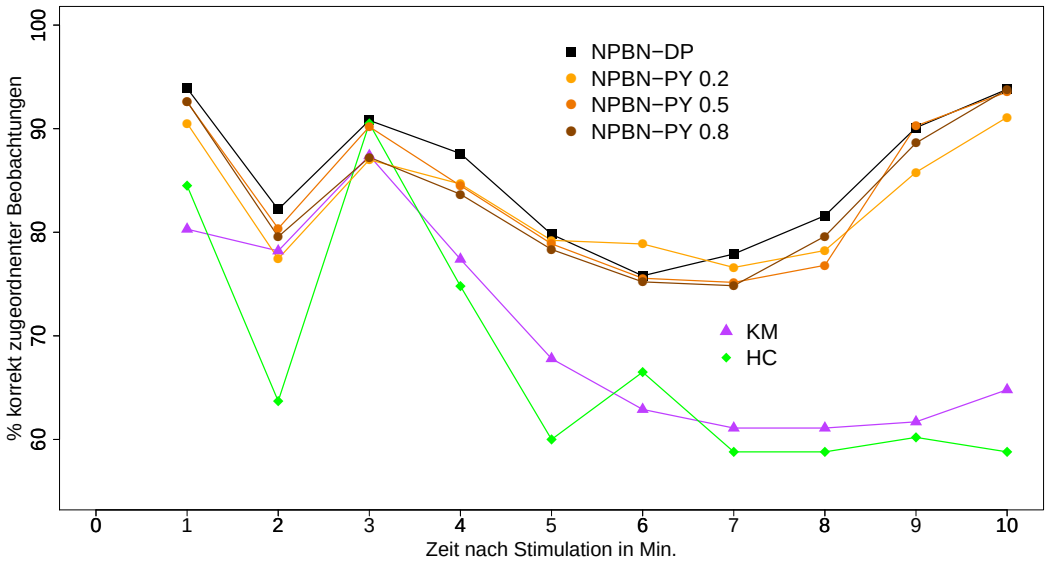


Abbildung 9: *Vergleich der Entmischungsgenauigkeit zwischen k-means (KM), hierarchischer Clusteranalyse (HC), nichtparametrischen Bayesschen Netzwerken mit der Dirichlet a priori Verteilung (NPBN-DP) sowie mit der Pitman-Yor a priori Verteilung mit verschiedenen Werten von σ (NPBN-PY02, NPBN-PY05, NPBN-PY08) bei einen Noisestärke von 0.7, aufgeschlüsselt nach der Zeit im Zwei-Komponenten-Fall.*

Die im vorigen Kapitel geäußerte Vermutung, dass NPBN-PY als Verallgemeinerung von NPBN-DP, welche die Anzahl der Komponenten mit berücksichtigt, in Datensätzen mit mehr Komponenten deutlich besser abschneidet, lässt sich hier nicht bestätigen. Es ist davon auszugehen, dass die Zahl der Komponenten im Datensatz mit vier noch nicht

hinreichend groß ist, um von den Eigenschaften des Pitman-Yor-Prozesses profitieren zu können. Abbildung 9 und 10 ab Seite 56 liefern für diese Behauptung einen weiteren Beleg.

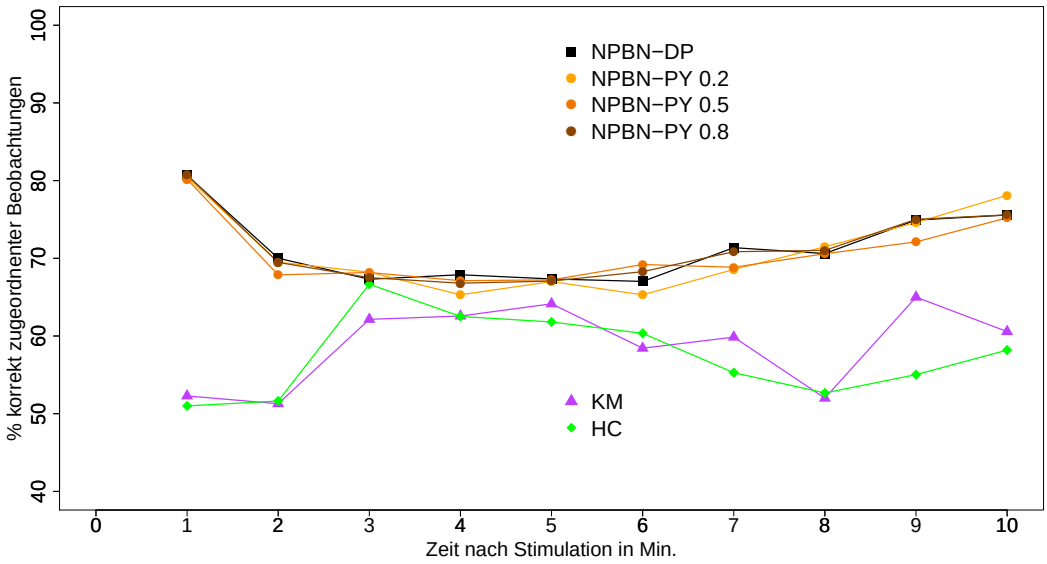


Abbildung 10: *Vergleich der Entmischungsgenauigkeit zwischen k-means (KM), hierarchischer Clusteranalyse (HC), nichtparametrischen Bayesschen Netzwerken mit der Dirichlet a priori Verteilung (NPBN-DP) sowie mit der Pitman-Yor a priori Verteilung mit verschiedenen Werten von σ (NPBN-PY02, NPBN-PY05, NPBN-PY08) bei einen Noisestärke von 0.7, aufgeschlüsselt nach der Zeit im Vier-Komponenten-Fall.*

Dort sind die *pco*-Werte für verschiedene Einstellungen der Parameter des Pitman-Yor-Prozesses für beide Fälle, zwei und vier Komponenten, gegenübergestellt. Für alle drei betrachteten Einstellungen des Pitman-Yor-Prozesses liegen die Ergebnisse nahe beieinander und dicht an dem *pco*-Wert des NPBN-Verfahrens mit dem Dirichlet-Prozess (welches identisch ist mit NPBN-PY $\sigma = 0$). Im Zwei-Komponenten-Datensatz erzielt NPBN-DP in einigen Fällen das beste Ergebnis (bei Min. 1), aber ebenso gibt es Fälle, in denen NPBN-PY mit $\sigma = 0.2$ (bei Min. 6), $\sigma = 0.5$ (bei Min. 9) oder $\sigma = 0.8$ (bei Min. 10) am besten abschneidet. Im Vier-Komponenten-Datensatz verhält es sich ähnlich, nur dass die Ergebnisse aller NPBN-Verfahren noch dichter zusammen liegen. Somit sind unabhängig von der Wahl des Parameters keine Anzeichen für eine generelle Überlegenheit von einem der beiden NPBN-Verfahren feststellbar.

4.6. Zusammenfassung

Insgesamt kann beobachtet werden, dass die Entmischung mittels der NPBN-Verfahren zufrieden stellend gelingt. Der simulierte Datensatz umfasst mehrere Szenarien. Neben verschiedenen Noisestärken werden auch unterschiedliche Netzwerktopologien, die sich über den Verlauf der Signalantwort manifestieren, darunter für die Entmischung günstige und auch ungünstige, berücksichtigt. Die erzielten Ergebnisse sind sowohl für die Reinheit der Gruppen als auch für die gefundene Komponentenzahl sehr gut. Die zum Vergleich herangezogenen klassischen Verfahren aus dem Bereich der Clusteranalyse schneiden schlechter ab. Die bessere Leistung der NPBN-Verfahren zeigt sich vor allem bei Konstellationen, in denen die Daten viele Subgruppen enthalten und eine hohe Noisestärke vorliegt. Ein Unterschied in der Leistungsfähigkeit zwischen NPBN-DP und NPBN-PY kann nicht festgestellt werden.

5. Schätzverfahren und Konzentrationsbestimmung am Beispiel des mating pathways in der Hefe

Die im Rahmen dieser Arbeit entwickelten Methoden sind, neben der Entmischung der Daten des Erk-Signalübertragungsnetzwerks aus Kapitel 4, in ein weiteres systembiologisches Projekt in Kooperation mit dem MPI Dortmund, Abteilung Systemische Zell-Biologie, eingeflossen, und haben die Analyse der erhobenen Daten entscheidend vorangebracht. In diesem Teil der Arbeit wird der Hintergrund und das Schätzverfahren zur Bestimmung der Konzentrationen der Proteinkomplexe erläutert. Die Auswertung der Schätzergebnisse und der darauf aufbauende Netzwerkinferenz wird im Kapitel 6 besprochen.

5.1. Problemstellung und Einordnung des Projekts

Ähnlich wie zuvor besteht die übergeordnete Fragestellung darin, die Interaktionsstruktur von Molekülen in biochemischen Netzwerken lebender Zellen aufzuklären. Am Modellorganismus Hefe (speziell an einem Derivat des Stamms S288c (Maglott et al., 2005)) soll der als mating pathway bekannte Stoffwechselweg erforscht werden, vergleiche Abbildung 11 auf Seite 60.

Eine der großen Schwierigkeiten dieses Vorhabens besteht darin, dass die Daten nicht in einer Form vorliegen, wie sie der NPBN-Algorithmus benötigt. Dieses Problem kann jedoch in einem Datenaufbereitungsschritt durch ein neu entwickeltes Schätzverfahren behoben werden (Jarzombek et al., 2014). Darauf wird im Abschnitt 5.3 genauer eingegangen.

Im Zentrum des im Projekt betrachteten biomolekularen Netzwerks stehen vier Proteine: Ste11 (systemischer Name YLR362W), Ste7 (systemischer Name YDL159W), Ste5 (systemischer Name YDR103W) und Fus3 (systemischer Name YBL016W). Diese können einzeln vorliegen oder untereinander Verbindungen eingehen und so Gebilde aus zwei (sogenannte Dimere), drei (sogenannte Trimere) und aus vier Proteinen (sogenannte Tetramere) entstehen lassen. Diese können aber wieder abgebaut werden, so dass sie für neue Verbindungen bereitstehen. Zur Vereinfachung werden die genannten Proteingebilde

sowie die einzelnen Proteine im Folgenden unter der Sammelbezeichnung Proteinkomplexe oder kurz Komplexe adressiert. Insgesamt sind 15 unterschiedliche Gebilde möglich, vergleiche Abbildung 3 auf Seite 70.

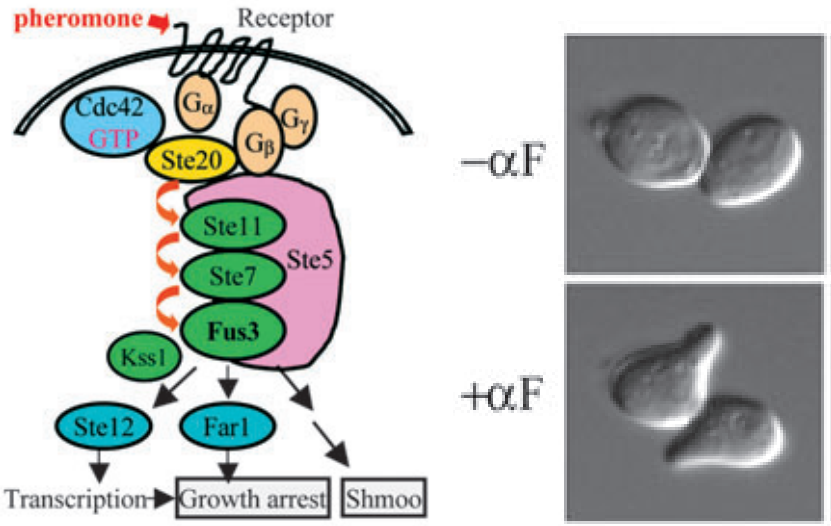


Abbildung 11: *Schematisches Diagramm des Hefe mating pathways (links). Nach der Bindung des α-Faktors an den Rezeptor wird eine Reaktion der Hefezelle ausgelöst, die vermittelt über mehrere Proteine, an zentraler Stelle über Ste7, Ste5, Ste11 und Fus3, zu der Bildung eines Shmoo führt. Unstimulierte Hefezelle (rechts oben). Hefezelle nach der Stimulation mit dem α-Faktor mit ausgebildeten Shmoo (rechts unten). Quelle: Qi und Elion (2005).*

Das spezielle Muster der Konzentrationen der Komplexe ist charakteristisch für die aktuellen Vorgänge in der Zelle und kann direkt Auskunft über den Zustand der Hefezelle geben. Zum Beispiel lässt sich erkennen, ob die Zelle kurz vor einer Teilung steht oder im Wachstum begriffen ist. Krankhafte Veränderungen oder andere Störungen des inneren Zellgleichgewichts werden hier ebenfalls sichtbar, was künftig zu neuen Diagnosemethoden führen könnte.

5.2. Versuchsaufbau und Messverfahren

Es ist bekannt, dass Hefezellen bei Stimulation mit einer bestimmten Konzentration eines bestimmten chemischen Botenstoffs ($10\mu M$ α-Faktor) reagieren, indem sie einen Shmoo, eine unter einem Lichtmikroskop sichtbare Ausstülpung der Zellmembran, ausbilden (Qi und Elion (2005), vergleiche Abbildung 11 auf Seite 60). Dies wirkt sich deutlich auf die inneren chemischen Prozesse aus. Der Zusammenhang ist gut belegt (Maeder et al., 2007) und wird häufig verwendet, um die Wirksamkeit neuer Untersuchungsmethoden zu zeigen. Die hier verwendete Experimentanordnung umfasst nicht stimulierten Zellen, im Folgenden abgekürzt mit US, kurz stimulierten Zellen (2-7 Min., SS) und lang stimulierten Zellen (40-90 Min., LS).

Die Konzentrationen der markierten Proteine werden mittels Fluoreszenzkorrelationsspektroskopie (FCS) erhoben, einer optischen Messmethode, mit der die Bestimmung selbst sehr niedriger molekularer Konzentrationen in einem flüssigen Medium möglich ist (Magde et al., 1972; Bacia et al., 2006). Das Verfahren ist eines der wenigen, das derartig präzise Messungen an lebenden Zellen erlaubt, und beliebig oft wiederholt werden kann. Es hat zudem den Vorteil, dass, wenn präparierte Zellen vorliegen, der Messaufwand verglichen mit anderen Verfahren gering ist, was unter moderaten Kosten größere Stichproben erlaubt.

Im Folgenden soll kurz auf die Besonderheiten des Messverfahrens eingegangen werden, um das Verständnis der Datenstruktur zu erleichtern. Durch Manipulation der Zell-DNA kann erreicht werden, dass neusynthetisierten Proteinen einer festgelegten Art ein zusätzliches Fluorophor anhaftet. Dieses hat die Eigenschaft, bei Anregung mit Licht einer spezifischen Wellenlänge selbst Licht einer bestimmten Wellenlänge zu emittieren (Shaner et al., 2007; Lippincott-Schwartz, 2003). Man sagt auch, dass das Protein markiert beziehungsweise, nach der englischen Bezeichnung, getagged wird. Aus der Intensität des gemessenen Lichtes lässt sich die Konzentration des markierten Proteins berechnen. Zur Erhebung der in diesem Projekt verwendeten Daten werden die Fluorophore 3-meGFP (grün) und 3m-Cherry (rot) verwendet. Im Rahmen einer Messung wird ein bestimmtes nanomolares Volumen in der Zelle für einige Sekunden fokussiert. Wann immer dieses von einem markierten Molekül passiert wird, wird eine Messung der emittierten Intensität und der Transitdauer vorge-

nommen. Daraus lässt sich die Konzentration aller Moleküle, welche genau ein markiertes Protein enthalten, ableiten. Diese wird abweichend vom Sprachgebrauch in der Statistik als „Autokorrelation“ bezeichnet und nach der englischen Schreibweise mit AC abgekürzt. Es ist ebenfalls möglich, die Konzentration aller Moleküle, welche beide markierten Proteine enthalten, zu bestimmen. Diese wird als „Kreuzkorrelation“ bezeichnet und mit CC abgekürzt. Insgesamt werden also pro Messung drei Werte erfasst: AC 1 (grün), AC 2 (rot) und CC, vergleiche auch Abbildung 12.

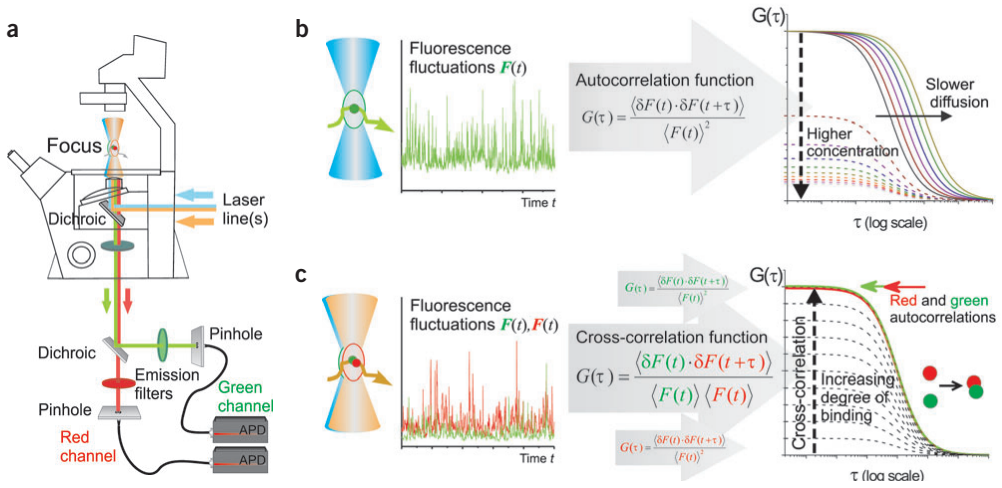


Abbildung 12: *Schematisches Diagramm des Verlaufs einer Konzentrationsmessung mittels Fluoreszenzkorrelationsspektroskopie. (a) Messaufbau, (b) einfarbiges FCS und (c) zweifarbiges FCS. Markierte Partikel diffundieren durch das Messvolumen und erzeugen ein Signal, welches registriert und in Konzentration umgerechnet wird. Quelle: Bacia et al. (2006).*

Da sich die Wellenlängen des Lichts zur Stimulation und die des emittierten Lichts deutlich unterscheiden müssen, gibt es nach heutigem Stand der Technik keine etablierte Möglichkeit mehr als zwei Proteine unterscheidbar zu markieren und zu messen. Es existieren zwar theoretische Überlegungen, wie man die Zahl der markierten Proteine auf drei erhöhen könnte, diese können zum aktuellen Zeitpunkt aber noch nicht praktisch umgesetzt werden. Die gängige Praxis ist es, unter Zuhilfenahme mehrerer möglichst gleichartiger Zellen alle paarweisen Kombinationen der Markierungen zu messen, und so zumindest eine grobe Übersicht der Konzentrationen in der Zelle zu erfassen (Maeder

et al., 2007). In dem für diese Arbeit relevanten Fall mit vier Proteinen sind die betrachteten Kombinationen: Ste11 Ste7, Ste11 Ste5, Ste11 Fus3, Ste7 Ste5, Ste7 Fus3, Ste5 Fus3, vergleiche auch Abbildung 13 auf Seite 63.

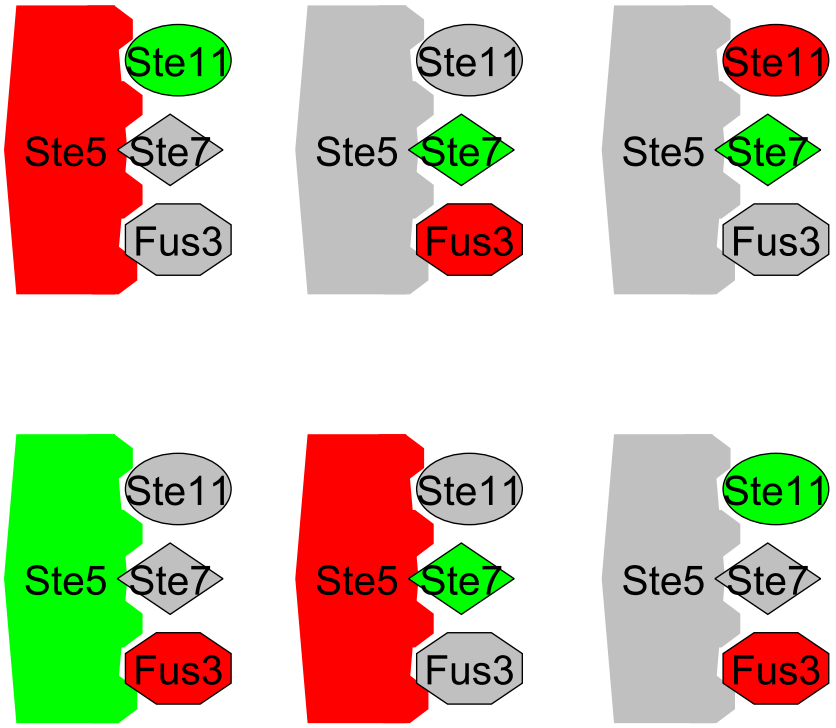


Abbildung 13: *Übersicht der sechs möglichen Kombinationen (Ste11 Ste5, Ste7 Fus3, Ste11 Ste7, Ste5 Fus3, Ste7 Ste5, Ste11 Fus3), anhand derer die vier Proteine (Ste11, Ste5, Ste7 und Fus3) mit zwei Farben markiert werden können.*

In bisherigen Arbeiten war es nicht möglich weiter vorzudringen, und alle Analysen konnten nur unter Vorbehalt stattfinden, da nicht klar war, ob hinter einer betrachteten AC Konzentration ein einzelnes Protein oder ein größeres Proteingebilde stand. So ist zum Beispiel im Falle, dass Ste7 und Fus3 markiert werden, allein anhand der FCS-Messung freies Ste7 (Notation: {Ste7}) und das Molekül bestehend aus Ste11 und Ste7 (Notation: {Ste11 Ste7}) nicht unterscheidbar (Maeder et al. (2007), vergleiche auch Abbildung 17 auf Seite 116). Dies ließe sich zwar mit weiteren Experimenten bestimmen, jedoch nur unter hohem zeitlichen und finanziellen Aufwand und mit der Konsequenz, dass das betrachtete biochemische System erheblich gestört wird, was die Ergebnisse wiederum verfälschen kann.

Die durch die Verfügbarkeit von nur zwei Markern bestehende Limitierung wird aber in Kauf genommen und von den Vorteilen mehr als aufgewogen.

5.3. Schätzverfahren zur Bestimmung der Konzentration der Proteinkomplexe

In diesem Abschnitt wird ein statistisches Verfahren vorgeschlagen, welches die technischen Beschränkungen des FCS-Konzepts aufweichen und dazu beitragen kann aus erhobenen Daten viel detailliertere Information zu gewinnen. Diese Schätzmethode, im Folgenden als Komplexeschätzer bezeichnet, erlaubt es beispielsweise sogenannte Dissoziationskonstanten für einzelne Proteine und beliebige Kombinationen dieser Proteine zu bestimmen und nicht mehr nur starre Gruppen (vergleiche Abbildung 17 auf Seite 116). Da diese Größe die Kinetik einer Reaktion charakterisiert, ist sie in der Systembiologie von besonderem Interesse. Beispielsweise lassen sich mit Hilfe von Dissoziationskonstanten die wahrscheinlichsten Zustände eines chemischen/biologischen Systems berechnen.

Im Folgenden wird der Komplexeschätzer für den verallgemeinerten Fall, in dem $\mathfrak{n}$ ($\mathfrak{n} \in \mathbb{N}$) Proteine vorliegen, beschrieben.

Gegeben seien die Proteine $\wp_1, \ldots, \wp_z, \ldots, \wp_\mathfrak{n}$, welche untereinander beliebig kombinierbar sind, jedoch nicht mehr als einmal in einem einzelnen molekularen Komplex vorkommen können. Die Komplexe werden mit $\mathscr{K}_\psi$, ihre unbekannten Konzentrationen mit $\mathcal{K}_\psi$ ($\psi = 1, \ldots, \sum_{z=1}^{\mathfrak{n}} \binom{\mathfrak{n}}{z}$) bezeichnet. Auf das Anwendungsbeispiel bezogen steht $\mathcal{K}_1$ für die Konzentration des Proteinkomplexes {Ste11} und $\mathcal{K}_{15}$ für die Konzentration des Proteinkomplexes {Ste11 Ste5 Ste7 Fus3}, vergleiche auch Abbildung 3 auf Seite 70. Deren Anzahl setzt sich aus $\binom{\mathfrak{n}}{1}$ freien Proteinen, $\binom{\mathfrak{n}}{2}$ Dimeren, $\binom{\mathfrak{n}}{3}$ Trimeren, $\binom{\mathfrak{n}}{4}$ Tetrameren usw. und einem Komplex, der aus allen $\mathfrak{n}$ Proteinen besteht, zusammen. Ein $\mathscr{K}_\psi$ lässt sich jeweils auch als Menge von Proteinen auffassen. Die Schreibweise $\wp_z \lhd \mathscr{K}_\psi$ bedeutet, dass das Protein $\wp_z$ in Komplex $\mathscr{K}_\psi$ enthalten ist. Analog bedeutet $\wp_z \not\lhd \mathscr{K}_\psi$, dass das Protein $\wp_z$ in dem betrachteten Komplex nicht vorkommt. verwendet.

Für jedes der $\binom{\mathfrak{n}}{2}$ möglichen Paare von markierten Proteinen ($\wp_z, \wp_{z^*}$, wobei o.B.d.A. $z < z^*$) sei y_{zz^*} der Vektor der in der korrespondierenden FCS-Messung ermittelten Konzentrationen bestehend aus den Autokorrelationswerten und dem Kreuzkorrelationswert der Proteine $\wp_z$ und $\wp_{z^*}$, $y_{zz^*} = \Big(\mathrm{AC}(\wp_z), \mathrm{AC}(\wp_{z^*}), \mathrm{CC}(\wp_z, \wp_{z^*})\Big)$. Die Anzahl der Messungen, bei denen simultan die Proteine $\wp_z$ und $\wp_{z^*}$ markiert werden, wird mit $\mathfrak{n}_{zz^*}$ bezeichnet.

Es wird davon ausgegangen, dass sich die gemessenen Werte additiv aus den Konzentrationen der beteiligten Komplexe zusammensetzen. Mögliche weitere biologische Einflüsse, wie zum Beispiel kompetitive Hemmung oder unterschiedliche Affinitäten, werden im Modell nicht berücksichtigt. Auf diesen Annahmen aufbauend soll für eine einzelne Messung y_{zz^*} folgendes Modell gelten:

$$y_{zz^*} = \begin{pmatrix} y_{zz^*}^{AC(\wp_z)} \\ y_{zz^*}^{AC(\wp_{z^*})} \\ y_{zz^*}^{CC(\wp_z\wp_{z^*})} \end{pmatrix} = \begin{pmatrix} \sum_{\psi\in\Lambda} \mathcal{K}_\psi + \varepsilon_z \\ \sum_{\psi\in\Delta} \mathcal{K}_\psi + \varepsilon_{z^*} \\ \sum_{\psi\in\Xi} \mathcal{K}_\psi + \varepsilon_{zz^*} \end{pmatrix}, \tag{17}$$

wobei $\Lambda := \{\mathcal{K}_\psi : \wp_z \lhd \mathscr{K}_\psi\}$, $\Delta := \{\mathcal{K}_\psi : \wp_{z^*} \lhd \mathscr{K}_\psi\}$, $\Xi := \{\mathcal{K}_\psi : \wp_z \lhd \mathscr{K}_\psi \wedge \wp_{z^*} \lhd \mathscr{K}_\psi\}$ und ε_z, ε_{z*} sowie ε_{zz^*} standardnormalverteilte Fehlerterme mit unbekannter Varianz repräsentieren. Abbildung 17 auf Seite 116 veranschaulicht die in Gleichung 17 beschriebene Zusammensetzung der Autokorrelationswerte und des Kreuzkorrelationswertes für das Beispiel des mating pathways mit $\wp_z$={Ste7} und $\wp_{z^*}$={Fus3}.

Das vorliegende Problem, die Bestimmung der Konzentrationen der einzelnen Komplexe, lässt sich trotz der Kenntnis des additiven Zusammenhangs nicht mathematisch exakt lösen. Für $\mathfrak{n} > 2$ übersteigt die Zahl der Unbekannten, also die Zahl aller möglichen Komplexe $\mathscr{K}_\psi$, $2^{\mathfrak{n}} - 1$, die Zahl der zur Verfügung stehenden unabhängigen Gleichungen, $\mathfrak{n} + \binom{\mathfrak{n}}{2}$. Das System ist also unterbestimmt. Es lässt sich aber als bedingte Regression, eingeschränkt auf die streng positiven reellen Zahlen, formulieren. Dies ermöglicht eine Schätzung der $\mathcal{K}_\psi$, welche jedoch aufgrund der Nichtidentifizierbarkeit des Modells starker Streuung unterliegen. Auf dieses Problem wird am Ende des folgenden Abschnitts noch eingegangen.

Das Regressionsproblem wird mit Hilfe der Designmatrix $\mathbf{X}$ formuliert, die sich anhand des in Gleichung (17) beschriebenen Zusammenhangs konstruieren lässt. Die

Spalten geben die Komplexe und die Zeilen die Messungen wieder. Jede Messung erweitert die Matrix um drei weitere Zeilen. Wird eine Messung hinzugefügt, entstehen die Zeilen $\mathbf{x}_I$, $\mathbf{x}_{II}$, $\mathbf{x}_{III}$, wobei in jeder Spalte ψ die Einträge $\mathbf{x}_{I\psi\cdots III\psi}$ von folgender Form sind:

$$\mathbf{x}_{I\psi} = \begin{cases} 1, & \wp_z \lhd \mathscr{K}_\psi \\ 0, & \wp_z \not\lhd \mathscr{K}_\psi \end{cases}, \; \mathbf{x}_{II\psi} = \begin{cases} 1, & \wp_{z^*} \lhd \mathscr{K}_\psi \\ 0, & \wp_{z^*} \not\lhd \mathscr{K}_\psi \end{cases}, \; \mathbf{x}_{III\psi} = \begin{cases} 1, & \wp_z \lhd \mathscr{K}_\psi \wedge \wp_{z^*} \lhd \mathscr{K}_\psi \\ 0, & \wp_z \not\lhd \mathscr{K}_\psi \vee \wp_{z^*} \not\lhd \mathscr{K}_\psi \end{cases}. \tag{18}$$

Eine Beispielmatrix für die Schätzung der Konzentrationen der Proteinkomplexe in einem System, das aus vier Proteinen hervorgeht, ist in Tabelle 7 auf Seite 124 abgebildet. Insgesamt besteht die Designmatrix $\mathbf{X}$ aus $2^{\mathfrak{n}} - 1$ Spalten und $3 \cdot \sum_{z=1}^{\mathfrak{n}-1} \sum_{z^*=z+1}^{\mathfrak{n}} \mathfrak{n}_{zz^*}$ Zeilen. Die Anzahl der Spalten entspricht somit der Länge des Vektors $\boldsymbol{\beta}$, der unbekannten $\mathscr{K}_w$. Die Anzahl der Zeilen entspricht der des Vektors $\mathbf{y}$, der FCS-Messungen y_{zz^*}, und der Länge des Vektors der zufälligen Fehler, genannt ε.

Damit lässt sich das betrachtete Regressionsproblem formulieren als:

$$\mathbf{y} = \boldsymbol{\beta} \cdot \mathbf{X} + \varepsilon \quad \text{mit} \quad \mathcal{K}_\psi \geq 0 \; \forall \psi. \tag{19}$$

Die gesuchte Größe kann durch das Lösen des Ausdrucks $\underset{\boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{X} \cdot \boldsymbol{\beta}\|_2$ bestimmt werden. Neben der L^2-Norm wird auch die L^1-Norm in Betracht gezogen. In Simulationsstudien hat sich aber gezeigt, dass die mit der L^1-Norm erzielten Schätzungen wesentlich stärker verzerrt sind und diese somit für das betrachtete Schätzproblem ungeeignet ist (Wolff, 2013).

Die Optimierung wird mit dem L-BFGS-B-Verfahren (Limited-memory-Broyden–Fletcher–Goldfarb–Shanno) von Byrd et al. (1995) durchgeführt, welches eine auf beschränkte Lösungsräume zugeschnittene Abwandlung des BFGS-Verfahrens (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970) darstellt. Ein solches Vorgehen ist in Fällen von nichtlinearen Optimierungsproblemen mit Nebenbedingungen gängig (Nocedal und Wright, 1999). Verwendet wird die in der Software R implementierte Funktion `optim`.

Durch die Einbettung des Schätzprozesses in ein Bootstrap-Gerüst (Efron, 1979) ist es möglich, die Verteilung der geschätzten Konzentrationen empirisch zu bestimmen. Dies erlaubt Rückschlüsse auf die Variabilität und die Stabilität der geschätzten Werte. So ist,

trotz des bestehenden Problems der Nichtidentifizierbarkeit eine im Sinne der Systembiologie verwertbare Quantifizierung der betrachteten Konzentrationen möglich (Efron und Tibshirani, 1993). Ein solches Vorgehen wurde bereits erfolgreich erprobt und wird in verschiedenen Publikationen diskutiert (Shotwell et al., 2013; Kirk und Stumpf, 2009). Eine Skizze der Implementierung ist im Anhang auf Seite 98 zu finden.

5.4. Anwendung des Komplexeschätzers auf die FCS-Messungen des Hefe mating pathways

Der hier untersuchte Datensatz besteht aus 862 FCS-Messungen der sechs möglichen Proteinpaare, gebildet aus den Proteinen Ste11, Ste5, Ste7 und Fus3, vergleiche Abbildung 13 auf Seite 63. Für eine umfassende Beschreibung der Datenstruktur und der Datenerhebung vergleiche Abschnitt 5.2. Die genaue Aufteilung der Messungen auf die Paare und auf die drei Stimulationsdauern kann aus Tabelle 2 entnommen werden.

markierte Proteine	nicht stimuliert (US)	kurz stimuliert (SS)	lang stimuliert (LS)
Ste7 Fus3	99	45	40
Ste11 Fus3	45	47	48
Ste11 Ste7	30	36	35
Ste11 Ste5	99	41	33
Fus3 Ste5	49	46	45
Ste5 Ste7	50	39	35
# FCS-Messungen	372	254	236
# erhobene Werte (AC1+AC2+CC)	1116	762	708

Tabelle 2: *Übersicht über die Datenstruktur der erhobenen FCS-Messungen an unstimulierter und stimulierter Hefe.*

Die Konzentrationen der Proteinkomplexe werden für jede der drei Experimentanordnungen (lang, kurz und nicht stimuliert) gesondert geschätzt. Es werden jeweils 1000

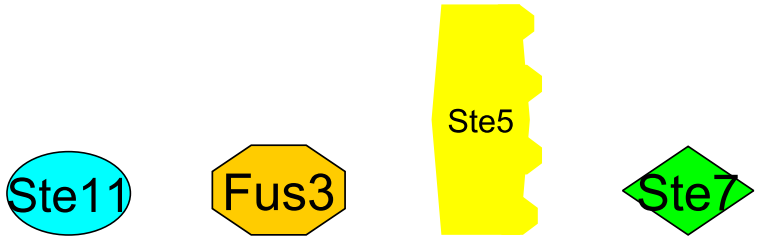
Bootstrap-Stichproben der Größe 167 erstellt. Das entspricht 501 Messwerten. Die Festlegung der Anzahl der Bootstrap-Stichproben und der Messwerte basiert auf Erfahrungswerten. Die Stichprobengröße ergibt sich aus dem Messverfahren (vergleiche Abschnitt 5.2), welches vorgibt, dass die 501 Messwerte auf drei Signale (AC_1, AC_2, CC) aufgeteilt werden müssen. Es wird dafür Sorge getragen, dass in jeder Bootstrap-Stichprobe alle der sechs möglichen Kombinationen der markierten Proteine gleich häufig vorkommen. Damit haben die Variablen aus Gleichung 19 auf Seite 66 folgende Dimension: $\boldsymbol{\beta} \in \mathbb{R}^{15\times1}, \mathbf{y} \in \mathbb{R}^{501\times1}, \varepsilon \in \mathbb{R}^{501\times1}$ und $\boldsymbol{X} \in \{0,1\}^{501\times15}$. Die Gestalt der Designmatrix $\boldsymbol{X}$ und der Vektoren $\mathbf{y}$ sowie $\boldsymbol{\beta}$ kann Tabelle 7 auf Seite 124 entnommen werden. Für die Schätzung werden die L^2-Norm und ein normalverteilter Fehler verwendet. Von den kombinatorisch 15 möglichen Komplexen, vergleiche Tabelle 3 auf Seite 70, können durch Expertenwissen zwei als nicht vorkommend identifiziert werden ($\mathscr{K}_7$: {Ste11 Ste7} und $\mathscr{K}_{12}$: {Ste11 Fus3 Ste7}). Ihre Konzentration wird auf null gesetzt, womit sich die Zahl der gesuchten Konzentrationen auf 13 reduziert ({Ste11}, {Fus3}, {Ste5}, {Ste7}, {Ste11 Fus3}, {Ste11 Ste5}, {Fus3 Ste5}, {Fus3 Ste7}, {Ste5 Ste7}, {Ste11 Fus3 Ste5}, {Ste11 Ste5 Ste7}, {Fus3 Ste5 Ste7 }, {Ste11 Fus3 Ste5 Ste7}). Das Gleichungssystem ist aber immer noch unterbestimmt. Die Schätzung erfolgt unter der Nebenbedingung, dass die Konzentration aller geschätzten Komplexe größer gleich null sein muss.

Vor der Auswertung werden die erhaltenen Werte gestutzt. Jeweils 20 % der größten und kleinsten Schätzungen in jeder der Bootstrap-Stichproben werden entfernt. Dieses Vorgehen ist aufgrund der hohen Streuung der Daten notwendig. Des Weiteren soll auf diese Weise der Tatsache Rechnung getragen werden, dass das System unterbestimmt ist. Indem die Ränder des Lösungsraumes abgeschnitten werden, wird die mögliche Verzerrung verkleinert.

Die Berechnung erfolgt mit der statistischen Programmiersprache R (R Development Core Team, 2013) unter Zuhilfenahme des Paketes `boot` (Canty und Ripley, 2011).

Monomere: Komplexe bestehend aus einem (freien) Protein

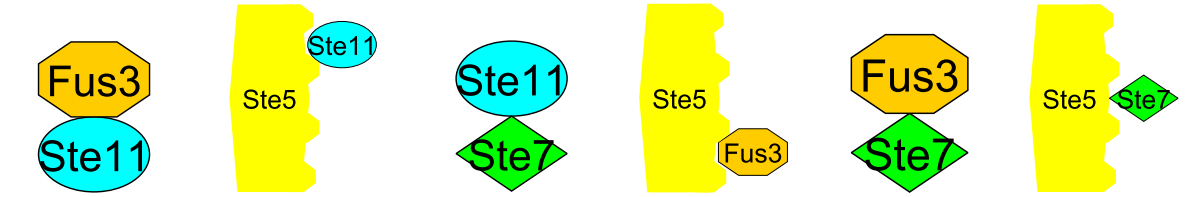
$\mathscr{K}_1$ $\mathscr{K}_2$ $\mathscr{K}_3$ $\mathscr{K}_4$



{Ste11} {Fus3} {Ste5} {Ste7}

Dimere: Komplexe bestehend aus zwei Proteinen

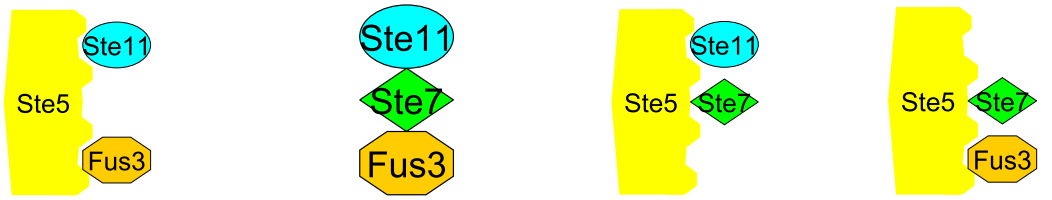
$\mathscr{K}_5$ $\mathscr{K}_6$ $\mathscr{K}_7$ $\mathscr{K}_8$ $\mathscr{K}_9$ $\mathscr{K}_{10}$



{Ste11 Fus3} {Ste11 Ste5} {Ste11 Ste7} {Fus3 Ste5} {Fus3 Ste7} {Ste5 Ste7}

Trimere: Komplexe bestehend aus drei Proteinen

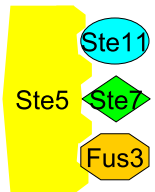
$\mathscr{K}_{11}$ $\mathscr{K}_{12}$ $\mathscr{K}_{13}$ $\mathscr{K}_{14}$



{Ste11 Fus3 Ste5} {Ste11 Fus3 Ste7} {Ste11 Ste5 Ste7} {Fus3 Ste5 Ste7}

Tetramer: Komplex bestehend aus vier Proteinen

$\mathscr{K}_{15}$



{Ste11 Fus3 Ste5 Ste7}

Tabelle 3: *Übersicht der betrachteten Komplexe des mating pathways der Hefe samt der zugehörigen Notation. Abgebildet sind die Schemata der vier zentralen Proteine Ste11, Fus3, Ste5 und Ste7, so genannte Monomere:* {*Ste11*}, {*Fus3*}, {*Ste5*}, {*Ste7*} *(1. Zeile) sowie alle 13 daraus ableitbaren Komplexe, so genannte Dimere:* {*Ste11 Fus3*}, {*Ste11 Ste5*}, {*Ste11 Ste7*}, {*Fus3 Ste5*}, {*Fus3 Ste7*}, {*Ste5 Ste7*} *(2. Zeile), Trimere:* {*Ste11 Fus3 Ste5*}, {*Ste11 Fus3 Ste7*}, {*Ste11 Ste5 Ste7*}, {*Fus3 Ste5 Ste7* } *(3. Zeile), Tetramer:* {*Ste11 Fus3 Ste5 Ste7*} *(4. Zeile).*

5.5. Diskussion

Die Genauigkeit der vom Komplexeschätzer produzierten Ergebnisse wurde, gesondert von der vorliegenden Arbeit, im Rahmen einer am Lehrstuhl für mathematische Statistik und biometrische Anwendungen der TU Dortmund, vom Verfasser der vorliegenden Arbeit betreuten, Bachelorarbeit untersucht (Wolff, 2013). Darin werden mehrere Variationen des Komplexeschätzers auf simulierten Datensätzen untersucht. Die in den Datensätzen abgebildeten Szenarios umfassen unterschiedliche Datenqualitäten (mit klarem und mit verzerrtem Signal) sowie Netzwerke unterschiedlicher Größe, darunter eines, welches von vier Proteinen erzeugt wird. Dieses entspricht in Umfang und Komplexität dem in der vorliegenden Arbeit untersuchten Problem. Gemäß der Ergebnisse der Bachelorarbeit erweist sich der Komplexeschätzer, gemessen am absoluten Fehler, als exakt und liefert auch bei verrauschten FCS-Messungen gute Schätzungen. Bei den deutlich größeren aus sechs Proteinen erzeugten Netzwerken sind die beobachteten Fehler jedoch geringfügig größer. Eine umfassendere Zusammenfassung der Bachelorarbeit und deren Ergebnisse befindet sich im Anhang im Abschnitt E. Die Praxistauglichkeit sowie die Eignung zur erfolgreichen Prozessierung von FCS-Messungen ist in Zusammenarbeit mit dem Max-Planck-Institut für molekulare Physiologie in Dortmund experimentell belegt worden (Jarzombek et al., 2014).

6. Auswertung der mittels des Komplexeschätzers aufbereiteten FCS-Messungen des Hefe mating pathways mit NPBN

In den folgenden drei Abschnitten werden zuerst die geschätzten Konzentrationen der Proteinkomplexe (6.1) sowie die Ergebnisse der Netzwerkschätzung (6.2) für getrennt vorliegende mating pathway Daten diskutiert. Der letzte Abschnitt (6.3) behandelt die Entmischungsergebnisse eines vermischten, aus den geschätzten Proteinkonzentrationen konstruierten Datensatzes.

6.1. Ergebnisse der Konstellationenschätzung

Eine in Zusammenarbeit mit Kooperationspartnern am Max-Planck-Institut für molekulare Physiologie in Dortmund durchgeführte, auf die biologischen Zusammenhänge ausgelegte Analyse der geschätzten Konzentrationen hat eine bislang nicht bekannte negative Rückkopplung aufgedeckt, bei der {Fus3} mit {Ste11} die Bildung der Shmoo reguliert. Neben der neuen Rückkopplung ist auch die Feststellung neu, dass {Fus3} und {Ste11} direkt miteinander interagieren. Dies weicht von der bisherigen Lehrmeinung, dass dies nur unter Mitwirkung von {Ste5}, einem sogenannten Gerüstprotein, möglich ist, ab. Beide Theorien wurden durch weitere gezielte Experimente bestätigt und stehen kurz vor der Publikation (Jarzombek et al., 2014). Die Ergebnisse der Schätzung der Proteinkonzentrationen sind in Abbildung 14 auf Seite 72 dargestellt. Die Schätzung erfolgt gemäß des in Abschnitt 5.3 vorgestellten Vorgehens. Die verwendeten Einstellungen des Komplexeschätzers sowie die Datengrundlage werden in Abschnitt 5.4 beschrieben.

Erste Hinweise auf die neu gefundenen Mechanismen findet man im Konzentrationsprofil von {Ste11 Fus3}. Kurz nach der Stimulation mit einem chemischen Botenstoff, dem α-Faktor, fällt die Konzentration, steigt aber nach längerer Zeit wieder an. Diese negative Rückkopplung hebt die Wirkung des α-Faktors auf und die Konzentration erreicht wieder das Niveau von nicht stimulierten Zellen. Dieses Verhalten lässt sich auch in den im An-

schluss betrachteten Netzwerken erkennen, welche aus den beschriebenen Daten geschätzt werden, vergleiche Abschnitt 6.2.

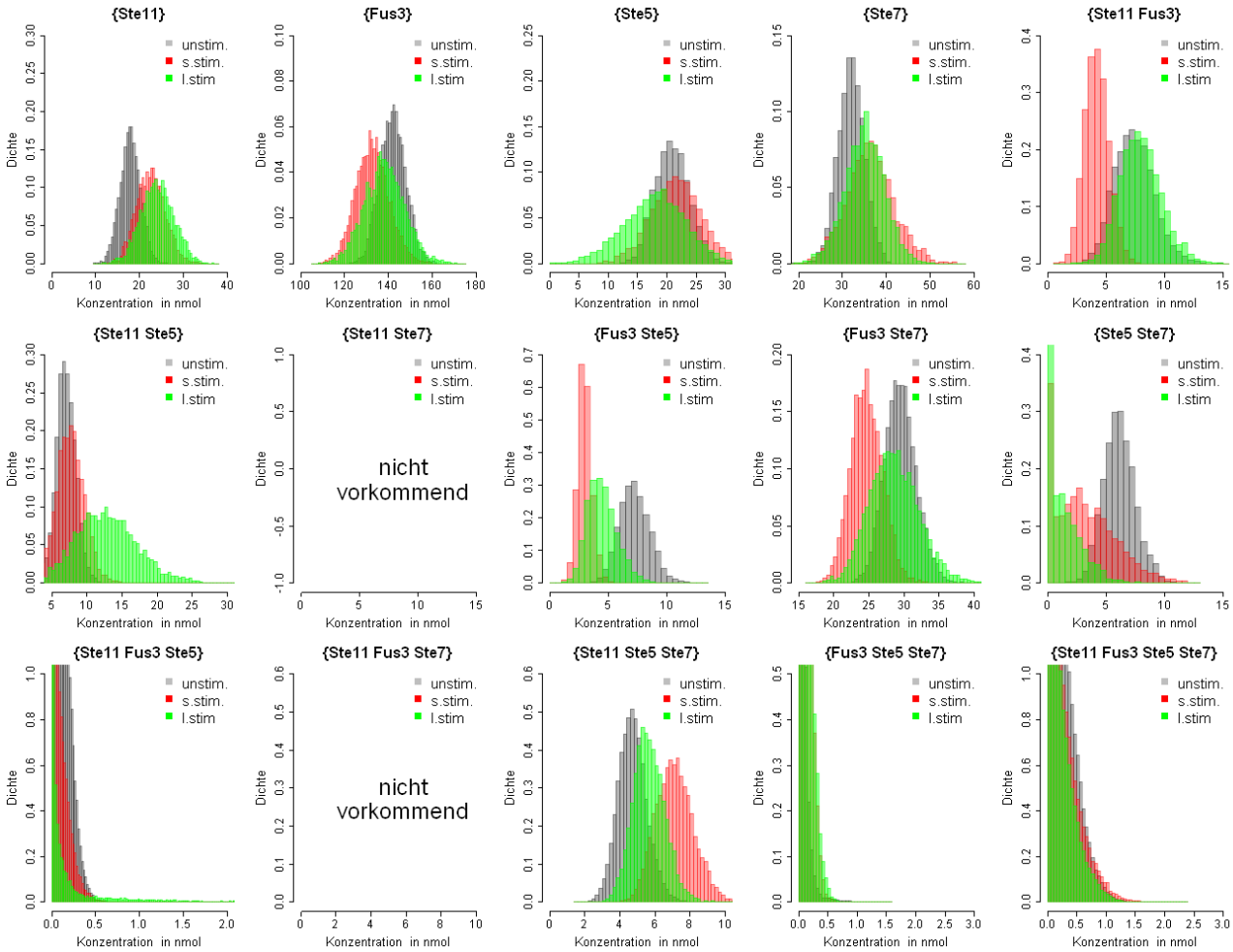
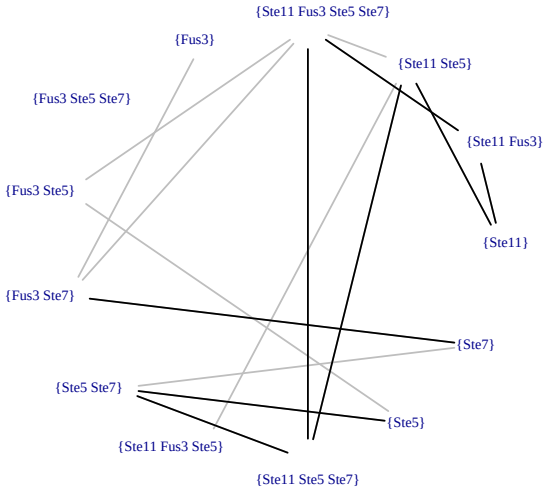
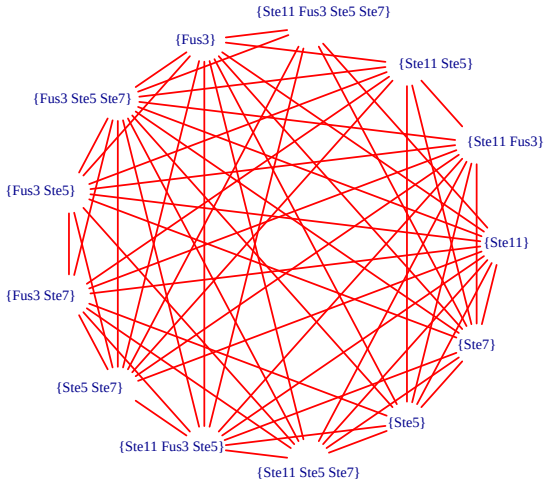


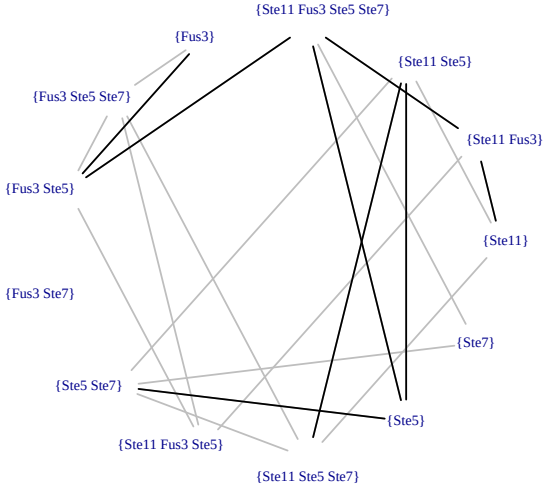
Abbildung 14: *Übersicht der geschätzten Verteilungen der Konzentrationen der Proteine, die im mating pathway der Hefe relevant sind ({Ste11},{Fus3}, {Ste5}, {Ste7}) und deren 11 Komplexe ({Ste11 Fus3}, {Ste11 Ste5}, {Ste11 Ste7}, {Fus3 Ste5}, {Fus3 Ste7}, {Ste5 Ste7}, {Ste11 Fus3 Ste5}, {Ste11 Fus3 Ste7}, {Ste11 Ste5 Ste7}, {Fus3 Ste5 Ste7 }, {Ste11 Fus3 Ste5 Ste7}). Abgebildet sind die Ergebnisse der Schätzung mit dem Komplexeschätzer. Als Grundlage dienen FCS-Messungen an Hefezellen, die nicht (grau), kurz (rot) und lang (grün) mit dem α-Faktor stimuliert wurden.*

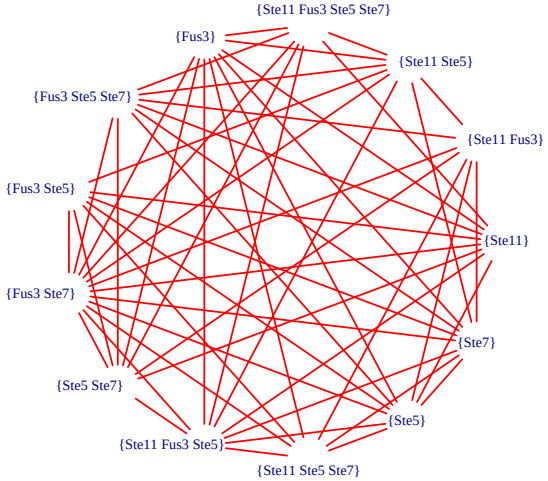
6.2. NPBN-Analyse der Konzentrationen der Proteinkomplexe

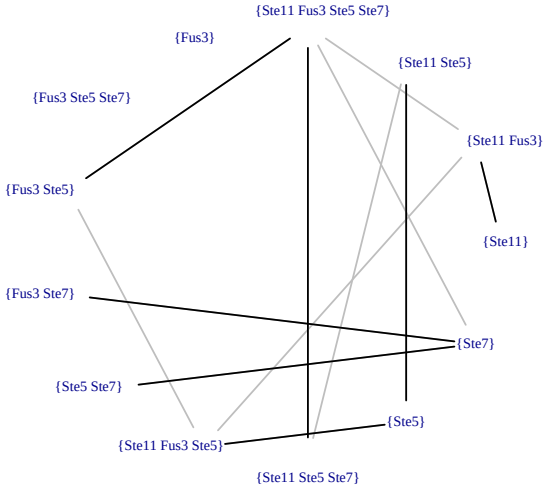
In diesem Abschnitt wird die Interaktionsstruktur zwischen den Komplexen des Hefe mating pathways betrachtet. Dazu werden die, im vorigen Abschnitt, mittels des Komplexeschätzers ermittelten Konzentrationen von nicht, kurz und lang stimulierter Hefe mit den Verfahren NPBN-DP (mit der mittels des Dirichlet-Prozesses erzeugten a priori Verteilung ($\theta = 1$)) und NPBN-PY (mit der mittels des Pitman-Yor-Prozesses erzeugten a priori Verteilung ($\theta = 1, \sigma = 0.2$)) ausgewertet. Verwendet wird jeweils ein Datensatz, welcher 175 geschätzte Konzentrationen der im mating pathway der Hefe relevanten Proteine ({Ste11}, {Fus3}, {Ste5}, {Ste7}) und deren 9 Komplexe ({Ste11 Fus3}, {Ste11 Ste5}, {Fus3 Ste5}, {Fus3 Ste7}, {Ste5 Ste7}, {Ste11 Fus3 Ste5}, {Ste11 Ste5 Ste7}, {Fus3 Ste5 Ste7 }, {Ste11 Fus3 Ste5 Ste7} umfasst. Die als nicht vorkommend eingestuften Komplexe {Ste11 Ste7} und {Ste11 Fus3 Ste7} wurden nicht berücksichtigt. Da die Daten getrennt vorliegen – es ist bekannt, welche Zellen wie lange mit dem α-Faktor stimuliert wurden – ist die Analyse der Ergebnisse auf die Struktur der Netzwerke ausgerichtet. Die vorgestellten Ergebnisse stammen aus MCMC-Simulationen mit $3.2 \cdot 10^6$ Iterationen mit einer Ausdünnung (thinning) von 350 und einem Burn-In von $2 \cdot 10^6$ Iterationen. Diese Einstellungen werden für alle drei betrachteten Zellen (nicht, kurz und lange stimulierte) und beide Varianten von NPBN verwendet. Die Schätzergebnisse von NPBN-DP und NPBN-PY liegen sehr nahe beieinander. Unterschiede sind, bis auf wenige Ausnahmen, erst ab der dritten Nachkommastelle der *pep*-Matrix vorhanden. Die zugehörigen geschätzten Kantenwahrscheinlichkeiten können aus den Abbildungen 18, 19 und 20 ab Seite 117 entnommen werden. Die aus den *pep*-Werten abgeleiteten Netzwerke sind für beide Varianten von NPBN identisch, so dass auf eine gesonderte Abbildung verzichtet wird, vergleiche Abbildung 15 auf Seite 75. In den dort abgebildeten Netzwerken können Hinweise auf einen Feedbackloop (Rückkopplung) ausgemacht werden. Zwischen den Komplexen {Ste11} und {Ste11 Fus3} gibt es für die 3 Stimulationsdauern eine starke Verbindung. Der korrespondierende Eintrag in der *pep*-Matrix ist in allen drei Fällen gleich eins. {Ste5} ist weder mit {Ste11} noch mit {Fus3} noch mit {Ste11 Fus3} verbunden. Die Einträge in der *pep*-Matrix liegen in den betreffenden Fällen bei maximal 0.32 und sonst deutlich tiefer, was ein starkes Indiz gegen das Vorliegen einer Verbindung zwischen diesen Knoten ist.


{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}


{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}


{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}


{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}


{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}

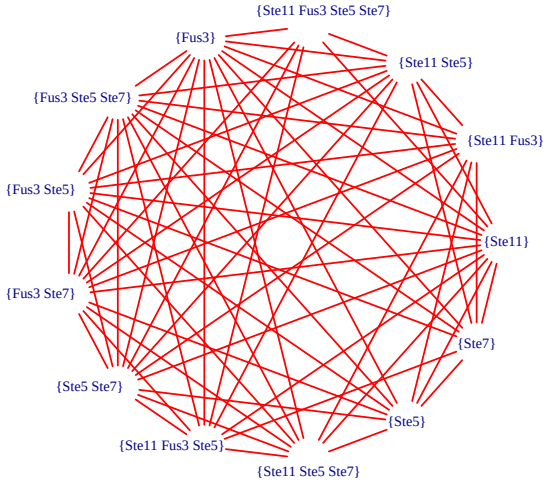

{Ste11 Fus3 Ste5 Ste7}
{Fus3}
{Ste11 Ste5}
{Fus3 Ste5 Ste7}
{Ste11 Fus3}
{Fus3 Ste5}
{Ste11}
{Fus3 Ste7}
{Ste7}
{Ste5 Ste7}
{Ste5}
{Ste11 Fus3 Ste5}
{Ste11 Ste5 Ste7}

Abbildung 15: *Netzwerke, geschätzt mit den NPBN-Schätzern (sowohl unter Verwendung der mit dem Dirichlet-Prozess erzeugten a priori Verteilung ($\theta = 1$) als auch unter Verwendung der mit dem Pitman-Yor-Prozess erzeugten a priori Verteilung ($\theta = 1, \sigma = 0.2$)). Da beide Varianten des Verfahrens das gleiche Netzwerk ermitteln, wurde auf eine gesonderte Abbildung verzichtet. Als Datengrundlage dienen die durch den Komplexeschätzer bestimmten Konzentrationen der im mating pathway der Hefe relevanten Proteine {Ste11}, {Fus3}, {Ste5}, {Ste7} und deren 9 Komplexe {Ste11 Fus3}, {Ste11 Ste5}, {Fus3 Ste5}, {Fus3 Ste7}, {Ste5 Ste7}, {Ste11 Fus3 Ste5}, {Ste11 Ste5 Ste7}, {Fus3 Ste5 Ste7 }, {Ste11 Fus3 Ste5 Ste7}. Die als nicht vorkommend eingestuften Komplexe {Ste11 Ste7} und {Ste11 Fus3 Ste7} werden nicht berücksichtigt. Berechnet für Hefezellen, welche mit dem α-Faktor nicht stimuliert (oben), kurz stimuliert (mittig) und lang stimuliert (unten) wurden. Für die Visualisierung der Ergebnisse der Netzwerkschätzung werden die folgenden Regeln angewandt: Kanten mit einem pep-Wert von 1 sind in schwarz, Kanten mit einem pep-Wert im Bereich von [0.6,1) in grau, und Kanten mit einem pep-Wert im Bereich von (0,0.4] rot eingezeichnet.*

Das Netzwerk, welches für die Zellen ohne Stimulation geschätzt wurde, ähnelt dem geschätzten Netzwerk lang stimulierter Zellen stärker als dem geschätzten Netzwerk kurz stimulierter Zellen. Die Zahl der übereinstimmenden Kanten dividiert durch die Summe der Kanten beider Netzwerke ergibt eine Ähnlichkeit von 0.36 zwischen den Netzwerken der nicht und der kurz stimulierten Hefe. Weiter werden Ähnlichkeiten von 0.38 zwischen den Netzwerken der nicht und der lang stimulierten sowie von 0.36 zwischen den Netzwerken der kurz und der lang stimulierten Hefe festgestellt. Dies stützt die Vermutung, dass die Stimulation der Hefezelle mit dem α-Faktor nach einer anfangs starken Reaktion mit der Zeit ihre Wirkung verliert und sich die Zelle ihrem Ausgangszustand nähert. Dieser Zusammenhang deutete sich bereits bei der Analyse der Proteinkomplexe im Abschnitt 6.1 an.

Eine alternative Möglichkeit, die Ähnlichkeit der gefundenen Graphen zu beurteilen, bietet der im Abschnitt 3.4 vorgestellte Distanzbegriff für DAGs.
Für die drei möglichen Paare der geschätzten Netzwerke (US-SS, US-LS sowie SS-LS) werden die Abstände berechnet und zwecks besserer Lesbarkeit auf eins normiert. Für die Netzwerke der nicht und der kurz stimulierten Hefe wird ein Abstand von 0.60 festgestellt, für die Netzwerke der nicht und der lang stimulierten Hefe ein Abstand von 0.16 und für die Netzwerke der kurz und der lang stimulierten Hefe wird ein Abstand von 0.60 berechnet.

Es besteht eine größere Ähnlichkeit zwischen den Netzwerken der nicht und der lang stimulierten Hefezellen, als zwischen den der nicht und kurz sowie zwischen den der kurz und lang stimulierten Hefezellen.

Die gefundenen Werte passen zu den bisherigen Ergebnissen und stimmen mit denen aus Abschnitt 6.1 überein. Die biologische Deutung des beobachteten Verhaltens lautet, dass die Zelle vor der Stimulation sich in einen Zustand des Gleichgewichtes befindet, welcher nach kurzer Stimulation aufgehoben wird, da die Zelle auf das empfangene chemische Signal reagiert. Nachdem die Reaktion stattgefunden hat, verliert der chemischer Botenstoff seine Wirkung auf die Zelle, so dass auch bei fortdauernder (langer) Stimulation die Zelle zu ihrem ursprünglichen Gleichgewicht zurückkehrt.

6.3. Entmischung künstlich vermischter Komplexekonzentrationen

Ergänzend zu der in Kapitel 4 durchführten Analyse der Leistungsfähigkeit von NPBN-DP und NPBN-PY bei der Entmischung wird diese auch bei dem deutlich größeren mating pathway betrachtet. Zu diesem Zweck wird der Datensatz der durch das Schätzverfahren aufbereiteten Messungen der Konzentrationen der Proteinkomplexe künstlich vermischt. Es werden zufällig jeweils 175 Beobachtungen der 13 Komplexekonzentrationen für nicht, kurz und lang stimulierte Zellen gezogen und zu einen Datensatz zusammengefügt. Anschließend wird dieser mit beiden Varianten des NPBN-Algorithmus ausgewertet. Verwendet werden die nichtparametrischen Bayesschen Netzwerke sowohl mit dem Dirichlet-Prozess $\theta = 1$ als auch mit den Pitman-Yor-Prozess $\theta = 1$, $\sigma = 0.2$ als a priori Verteilung. Die vorgestellten Ergebnisse stammen aus MCMC-Simulationen mit $3.2 \cdot 10^6$ Iterationen mit einer Ausdünnung (thinning) von 500 und einem Burn-In von $2 \cdot 10^6$ Iterationen.

Die Entmischung funktioniert für beide Varianten des Verfahrens gut. Für die Analysen unter Verwendung der mit dem Dirichlet-Prozess erzeugten a priori Verteilung ergibt sich ein *pco*-Wert von 96.75 %, für die unter Verwendung der mit dem Pitman-Yor-Prozess erzeugten a priori Verteilung ein *pco*-Wert von 91.80 %. Das um ca. 5 % bessere Ergebnis bei Verwendung der Dirichlet a priori Verteilung unterstützt zusätzlich die im Abschnitt 4.5 formulierte Vermutung, dass die geringe Zahl an Subgruppen und somit nur wenigen Komponenten den Dirichlet-Prozess begünstigt, da der Pitman-Yor-Prozess darauf ausgelegt ist, wenige große und viele kleine Komponenten zu generieren. Es lässt sich jedoch keine definitive Aussage über die unterschiedliche Leistungsfähigkeit der Verfahren treffen, da beide nicht unerheblichen Schwankungen unterliegen. Dies konnte bereits im Kapitel 4 festgestellt werden, vergleiche Abbildungen 5 und 6 auf Seite 48 und 49.

6.4. Zusammenfassung

Der entwickelte Ansatz zur Schätzung der Konzentration der Proteinkomplexe zeigt in der Anwendung gute Ergebnisse. Die in der Hefezelle gemessenen Werte können erstmals so aufbereitet werden, dass die Erstellung eines vollständigen Bildes des betrachteten Systems möglich ist. Die mittels NPBN durchgeführte Rekonstruktion der Interaktionsmuster zwischen den Proteinkomplexen gelingt auch bei im Rahmen von Experimenten erhobenen Labordaten mit zufriedenstellender Qualität. Für biochemische Netzwerke, von denen aufgrund der experimentellen Anordnung Unterschiede bekannt sind, weisen die Schätzungen eine unterschiedliche, biologisch plausible Beschaffenheit auf. Aus den rekonstruierten Netzwerken kann ein bislang unbekannter biologischer Zusammenhang abgeleitet werden. Es kann ebenfalls beobachtet werden, dass die Entmischung der vermischten Datensätze erfolgreich verläuft. Es werden *pco*-Werte deutlich über 90 % erreicht.

7. Diskussion und Ausblick

Die Frage, wie einzelne Beobachtungen zusammenhängen und wie aus Informationen über einzelne Teile auf ihr Zusammenwirken und schlussendlich auf das Funktionieren des übergeordneten Systems geschlossen werden kann, ist für viele Forschungsbemühungen von zentraler Bedeutung. Für ihre Beantwortung hat sich das Konzept der Netzwerkinferenz als wirksames Werkzeug erwiesen und kommt vor allem im Bereich der Systembiologie vermehrt zur Anwendung. Hier werden die Interaktionsmuster der zahllosen biochemischen Akteure auf diese Weise untersucht. Auch außerhalb des akademischen Umfeldes sind die Vorteile des Netzverkansatzes erkannt worden und werden in Industrie und Wirtschaft zunehmend geschätzt und genutzt.

In dieser Arbeit werden neue statistische Konzepte zur Erkennung und Analyse von Interaktionsmustern vorgestellt. Diese werden sowohl an simulierten Daten aus dem Erk-Signalübertragungsnetzwerk als auch an experimentellen Daten des mating pathways der Hefe mit Erfolg zur Anwendung gebracht. Obwohl der Fokus hier auf der Anwendung in biochemischen Proteinnetzwerken liegt, sind die vorgeschlagenen Methoden auch abseits der Systembiologie anwendbar.

Methodisch kann die Arbeit in zwei Themenschwerpunkte eingeteilt werden. Den Hauptschwerpunkt bildet das aus den Bayesschen Netzwerken entwickelte Verfahren der nichtparametrischen Bayesschen Netzwerke. Dieses ist, so weit bekannt, als einzige Netzwerkinferenzmethode in der Lage, Subgruppen innerhalb der Daten zu erkennen und die Beobachtungen zu partitionieren. Weiter gelingt es in dieser Arbeit, neben dem Dirichlet-Prozess den Pitman-Yor-Prozess als a priori Verteilung der Clusterstruktur zu adaptieren. Beide Varianten des Verfahrens werden bezüglich ihrer Leistungsfähigkeit bei der Entmischung von Beobachtungen untersucht. In allen betrachteten Fällen sind sie den verfügbaren Alternativen überlegen. Bei einem Vergleich der Entmischungsqualität unter verschiedenen a priori Verteilungen kann kein relevanter Unterschied gefunden werden.

Ein interessanter Aspekt, den man auf dieser Arbeit aufbauend betrachten könnte, ist das Verhalten der nichtparametrischen Bayesschen Netzwerke bei Vorliegen einer Gruppenstruktur, in der die Subgruppen nicht gleich groß sind. Bei dieser Fragestellung sollte

erörtert werden, wie klein eine Subgruppe werden kann, bevor sie nicht mehr als eigenständig eingestuft wird. Ebenso könnte ein Datensatz mit sehr vielen Subgruppen untersucht werden, um zu prüfen, ob sich dann eine der a priori Verteilungen als überlegen erweist. Diese Betrachtung sollte für unterschiedliche Noisestärken unternommen werden. Erste Arbeiten zur Beantwortung dieser Fragen liegen bereits vor, siehe Wieczorek et al. (2015). Untersucht wird ein Netzwerk mit zwei Komponenten, welche im Verhältnis 9 zu 1 vorliegen. Für geringes Noiseniveau (0.2) sind die mit dem NPBN-DP Ansatz erzielten Klassifikationsergebnisse vergleichbar mit dem in dieser Arbeit untersuchten 1 zu 1 Fall. Für ein hohes Niveau (0.7) werden die NPBN-DP Ergebnisse schlechter und liegen auf dem Niveau von k-means, aber über dem von HC.

Weiter erscheint es lohnenswert zu untersuchen, ob die guten Ergebnisse des NPBN durch Kombination mit komplementären Ansätzen im Rahmen eines ensemble Konzepts noch weiter verbessert werden können, wie beispielsweise in der Arbeit von Marbach et al. (2010) oder kürzlich von Meyer et al. (2014) vorgeschlagen. So könnte ein Datensatz mittels NPBN entmischt und die Abhängigkeitsstruktur der gefundenen Netzwerke bestimmt werden. Im Anschluss daran könnten Petri Netze eingesetzt werden, um die Art der Interaktion durch Schätzung der Kantengewichte zu spezifizieren. Ebenso könnte eine „schnelle“, dafür aber nur begrenzt leistungsfähige Netzwerkinferenzmethode, wie z.B. die Methoden aus Abschnitt 2.2, eingesetzt werden, um informative Startwerte für das NPBN-Verfahren zu bestimmen und so dessen Burn-In-Phase zu verkürzen.

Eine Idee, wie die hier gewonnenen Erkenntnisse für neue Probleme und somit einen größeren Anwenderkreis nutzbar gemacht werden können, stellt die Übertragung auf Fragestellungen mit räumlichen und/oder zeitlichen Aspekten dar. Für letztere existiert bereits ein theoretisches Fundament, die Dynamischen Bayesschen Netzwerke (Ghahramani, 1998). Diese erlauben die Eingabe von Zeitreihen und ermöglichen es damit die Interaktion der Knoten im zeitlichen Verlauf zu untersuchen sowie die sogenannten self loops, den Einfluss einer Variablen auf sich selbst, zu berücksichtigen. Einen ersten Ansatz bei der Einbettung räumlicher Informationen in Bayessche Netzwerken könnte eine entsprechende Modifikation der a priori Verteilung bei der Bestimmung der Verbindungen (single edge operations) darstellen. Kanten zwischen räumlich nahen Knoten könnten eine höhere a priori Wahr-

scheinlichkeit, vorgeschlagen und akzeptiert zu werden, zugewiesen bekommen als jene zwischen entfernten Knoten.

Für den breiten Einsatz von NPBN bei größeren Netzwerken, mit deutlich mehr als 50 Knoten, ist es wünschenswert die Rechenzeit zu verringern. Dies könnte durch eine effizientere Implementierung in schnellen Sprachen wie zum Beispiel C++ umgesetzt werden. Ferner könnte die Faktorisierbarkeit der Likelihood der Allokationsstruktur ($\boldsymbol{l}$) und die der Graphenstruktur ($\mathcal{G}$) ausgenutzt werden, indem nur die Faktoren neu berechnet werden, die in der gegebenen Iteration verändert wurden, während die restlichen wiederverwendet werden. Diese Maßnahme wäre umso wirkungsvoller, je mehr Knoten und je mehr Komponenten das betrachtete Netzwerk enthält. Ein weiterer Punkt, der die Leistungsfähigkeit des NPBN-Algorithmus verbessern könnte, betrifft die Operationen am Allokationsvektor. Im Teilungsschritt werden die Beobachtungen zur Befüllung der neuen Komponente zufällig ausgewählt. Hierbei besteht die Gefahr, dass zusammengehörende Beobachtungen getrennt werden. Dies macht den Schritt ineffizient und den Algorithmus langsamer. Eine gezieltere Auswahl der für die Befüllung ausgesuchten Beobachtungen würde diese Mängel beheben.

Den zweiten Schwerpunkt der Arbeit bildet die Entwicklung einer Methode zur Schätzung von Proteinkonzentrationen, dem Komplexeschätzer. Mit ihm ist es möglich, aus Fluoreszenzkorrelationsspektroskopiemessungen (FCS) nicht wie bisher nur feste Gruppen von Proteinen zu quantifizieren, sondern gezielt einzelne Proteine und beliebige vom Anwender ausgewählte Gruppen von Proteinen zu bestimmen. Dies stellt eine deutliche Verbesserung zum gegenwärtigen Standard dar und erhöht den Informationsgewinn durch FCS-Messungen entscheidend. Es ist nun erstmals möglich, die gemeinsame Konzentration aller erfassten Objekte in einer Zelle zu schätzen. Mit Hilfe dieser Methode konnte eine in der Biologie bisher unbekannte Rückkopplung im Hefe mating pathway gefunden werden.

Die Leistungsfähigkeit des vorgestellten Komplexeschätzers kann durch geringe Modifikationen weiter verbessert werden. Für die Umsetzung ist es jedoch erforderlich, dass abweichend von der bis jetzt üblichen Verfahrensweise mehrere der betrachteten Proteine identisch markiert werden. Die notwendigen zusätzlichen Präparationen der Zellen sowie die erneute Messung konnten im Rahmen dieser Arbeit jedoch nicht realisiert werden.

Dieser Ansatz sollte aber unbedingt weiterverfolgt werden, da hiermit das Potential der FCS-Messungen noch weiter gesteigert werden kann, so dass auch große, viele Proteine umfassende Systeme in hoher Auflösung analysiert werden können. Dies ist nach heutigem Stand der Technik mit durch die nur zwei verfügbaren Markierungen stark begrenzter Messgenauigkeit nicht möglich.

Theoretisch bietet die Mehrfachmarkierung in Verbindung mit geeigneten Versuchsplänen und dem Komplexeschätzer die Möglichkeit mit wenigen Experimenten Ergebnisse zu erhalten, die gleichwertig sind zu jenen, die mit mehreren Experimenten bei einfach markierten Proben erzielt werden. Gegenwärtig ist nicht ersichtlich, ob die eingesparten Messungen den Aufwand der Mehrfachmarkierung aufwiegen. Diese Flexibilisierung der Messtechnik bietet aber zusätzliche Optionen, z.B. können mit gleicher Anzahl Messgeräte mehr Proben untersucht werden, was bei beschränkten Messkapazitäten einen großen Vorteil darstellt.

Im Rahmen der Arbeit wird auch ein Konzept zum Clustern von gerichteten azyklischen Graphen (DAGs) entwickelt. Im Gegensatz zu den in der Literatur vorgeschlagenen Verfahren werden an die Daten keine speziellen Anforderungen gestellt. Es müssen lediglich DAGs eines festen Zeitpunkts verwendet werden. Konkret wird ein Distanzbegriff für DAGs entwickelt, welcher die Eigenschaften einer Semimetrik erfüllt. Mit ihm ist es möglich eine sinnvolle Ähnlichkeitsmatrix aufzustellen, welche zum Clustern benutzt werden kann. Dies erlaubt es Netzwerke, welche gerade erst geschätzt wurden, direkt zum aktuellen Wissensstand in Bezug zu setzen. Darauf aufbauend könnten künftig neue Diagnoseverfahren konzipiert werden, die an Patienten erhobene Netzwerke mit Referenznetzwerken Kranker und Gesunder abgleichen. Die Limitierung des Clusterverfahrens auf statische Netzwerke könnte durch eine Modifikation der Abstandsfunktion $\mathcal{F}$ aufgehoben werden. Als Ausgangspunkt erscheint die marginale Likelihoodfunktion der dynamischen Bayesschen Netzwerke sehr geeignet.

Insgesamt konnte mit dieser Arbeit der “Werkzeugkasten“ der Systembiologie um mehrere wertvolle Methoden ergänzt werden, von denen künftige Forschungsbemühungen auf diesem Gebiet profitieren können.

Literatur

Altay, G., M. Asim, F. Markowetz und D. Neal (2011). Differential C3NET reveals disease networks of direct physical interactions. *BMC Bioinformatics 12*.

Altay, G. und F. Emmert-Streib (2010). Revealing differences in gene network inference algorithms on the network level by ensemble methods. *Bioinformatics 26*(14), 1738–1744.

Antoniak, C. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Annals of Statistics 2*, 1152–1174.

Armbruster, D., P. Degond und C. Ringhofer (2005). Continuum models for interacting machines. In D. Armbruster, K. Kaneko und A. Mikhailov (Hrsg.), *Networks of interacting machines*. World Scientific Publishing, Singapore.

Bacia, K., S. Kim und P. Schwille (2006). Fluorescence cross-correlation spectroscopy in living cells. *Nature Methods 3*(2), 83–89.

Banerjee, O., L. El Ghaoui und A. d'Aspremont (2008, March). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research 9*, 485–516.

Basso, K., A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera und A. Califano (2005). Reverse engineering of regulatory networks in human B cells. *Nature Genetics 37*(4), 382–390.

Becker, T., M. Meyer, M. E. Beber, K. Windt und M.-T. Hütt (2013). A comparison of network characteristics in metabolic and manufacturing systems. In H.-J. Kreowski, B. Scholz-Reiter und K.-D. Thoben (Hrsg.), *Dynamics in Logistics: Third International Conference, LDIC 2012 Bremen, Germany, February/March 2012 Proceedings*. Springer.

Berestovsky, N., W. Zhou, D. Nagrath und L. Nakhleh (2013). Modeling integrated cellular machinery using hybrid Petri-Boolean networks. *PLoS Computational Biology 9(11)*.

Blackwell, D. und J. B. MacQueen (1973). Ferguson distributions via Pólya urn schemes. *Annals of Statistics 1*, 353–355.

Brandes, U. und T. Erlebach (Hrsg.) (2005). *Network Analysis*, Volume 3418 of *Lecture Notes in Computer Science*. Spinger.

Brooks, S. P. und A. Gelman (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics 7*(4), 434–455.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms. *Journal of the Institute of Mathematics and Its Applications. 6*, 76–90.

Butte, A. und I. Kohane (2000). Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. *Pacific Symposium on Biocomputing*, 418–429.

Byrd, R., P. Lu, J. Nocedal und C. Zhu (1995). A limited memory algorithm for bound constrained optimization. *SIAM J. Scientific Computing 16*, 1190–1208.

Canty, A. und B. Ripley (2011). *boot: Bootstrap R (S-Plus) Functions*. R package version 1.3-2.

Chaouiya, C. (2007). Petri net modelling of biological networks. *Briefings in Bioinformatics 8*(4), 210–219.

Chen, T., H. He und G. Church (1999). Modeling gene expression with differential equations. *Pacific Symposium on Biocomputing 4*, 29–40.

Cooper, G. F. und E. Herskovits (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning 9*, 309–347.

Davis, R. J. (1993). The mitogen-activated protein kinase signal transduction pathway. *Journal of Biological Chemistry 268*, 14553–14553.

de Jong H. (2002). Modeling and simulation of genetic regulatory systems: a literature review. *Journal of Computational Biology 9 (1)*, 67–103.

de Smet, R. und K. Marchal (2010). Advantages and limitations of current network inference methods. *Nature Reviews Microbiology 8 (10)*.

Di Fiore, P. P. und G. N. Gillt (1999). Endocytosis and mitogenic signaling. *Current opinion in cell biology 11*(4), 483–488.

Diestel, R. (2010). *Graphentheorie*. Springer, Berlin.

Donghyeon, Y., K. MinSoo, X. Guanghua und H. Tae Hyun (2013). Review of biological network data and its applications. *Genomics & Informatics 11(4)*, 200–210.

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *The Annals of Statistics 7*, 1–26.

Efron, B. und R. Tibshirani (1993). *An Introduction to the Bootstrap*. Chapman & Hall.

Egan, S. E., B. W. Giddings, M. W. Brooks, L. Buday, A. M. Sizeland und R. A. Weinberg (1993). Association of Sos Ras exchange protein with Grb2 is implicated in tyrosine kinase signal transduction and transformation. *Nature 363*, 45–51.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics 1*, 209–230.

Fletcher, R. (1970). A new approach to variable metric algorithms. *Computer Journal. 13*, 317–322.

Friedman, N. und D. Koller (2003). Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning 50*, 95–125.

Friedman, N., M. Linial, I. Nachman und D. Pe'er (2000). Using Bayesian networks to analyze expression data. *Journal of Computational Biology 7*, 601–620.

Friedman, N., K. Murphy und S. Russell (1998). Learning the structure of dynamic probabilistic networks. 139–147. Morgan Kaufmann.

Fritsch, A. (2009). *mcclust: Process an MCMC Sample of Clusterings*. R package version 1.0.

Fritsch, A. und K. Ickstadt (2009). Improved criteria for clustering based on the posterior similarity matrix. *Bayesian Analysis 4*, 367–392.

Gale, N. W., S. Kaplan, E. J. Lowenstein, J. Schlessinger und D. Bar-Sagi (1993). Grb2 mediates the EGF-dependent activation of guanine nucleotide exchange on Ras. *Nature 363*, 88–92.

Geiger, D. und D. Heckerman (1994). Learning Gaussian networks. In R. L. de Mántaras und D. Poole (Hrsg.), *Uncertainty in Artificial Intelligence Proceedings of the Tenth Conference*, 235–243.

Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In J. M. Bernardo, J. O. Berger, A. P. Dawid und A. F. M. Smith (Hrsg.), *Bayesian Statistics 4.*, The Valencia Meetings. Oxford University Press.

Ghahramani, Z. (1998). Learning dynamic Bayesian networks. In C. Giles und M. Gori (Hrsg.), *Adaptive Processing of Sequences and Data Structures*, Volume 1387 of *Lecture Notes in Computer Science*, 168–197. Berlin: Springer.

Ghosh, J. K. und R. V. Ramamoorthi (2003). *Bayesian Nonparametrics*. New York: Springer.

Gill, R., S. Datta und S. Datta (2010). A statistical framework for differential network analysis from microarray data. *BMC Bioinformatics*.

Giudici, P. und R. Castelo (2003). Improving Markov chain Monte Carlo model search for data mining. *Machine Learning 50*(1-2), 127–158.

Goldfarb, D. (1970). A family of variable metric updates derived by variational means. *Mathematics of Computation. 24*, 23–26.

Gotoh, Y., E. Nishida, T. Yamashita, M. Hoshi, M. Kawakami und H. Sakai (1990). Microtubule-associated-protein (MAP) kinase activated by nerve growth factor and epidermal growth factor in PC12 cells. *European journal of biochemistry 193*(3), 661–669.

Grzegorczyk, M. (2011). Bayesian networks for reconstructing gene regulatory networks in systems biology research. Habilitationsschrift, Fakultät Statistik, Technische Universität Dortmund.

Grzegorczyk, M. und D. Husmeier (2008). Improving the structure MCMC sampler for Bayesian networks by introducing a new edge reversal move. *Machine Learning 71*, 265–305.

Grzegorczyk, M., D. Husmeier, K. D. Edwards, P. Ghazal und A. J. Millar (2008). Modelling non-stationary gene regulatory processes with a non-homogeneous Bayesian network and the allocation sampler. *Bioinformatics 24*(18), 2071–2078.

Hardy, S. und P. N. Robillard (2008). Petri net-based method for the analysis of the dynamics of signal propagation in signaling pathways. *Bioinformatics 24*(2), 209–217.

Hasenauer, J., H. C., H. T. und T. FJ. (2014). ODE constrained mixture modelling: a method for unraveling subpopulation structures and dynamics. *PLoS Comput Biol. 10*(7).

Hjort, N. L., C. Holmes, P. Müller und S. G. Walker (Hrsg.) (2010). *Bayesian Nonparametrics*. Cambridge: Cambridge University Press.

Hoerl, A. E. und R. W. Kennard (2000). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics 42*(1), 80–86.

Ickstadt, K., B. Bornkamp, M. Grzegorczyk, J. Wieczorek, G. H. E. Sherriff, M. R. und E. Zamir (2011). Nonparametric Bayesian networks (with discussion). In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith und M. West (Hrsg.), *Bayesian Statistics 9.*, The Valencia Meetings, 283–316. Oxford University Press.

Jackson, M. O. (2010). *Social and Economic Networks*. Princeton University Press.

Jain, A. K. und R. Dubes (1988). *Algorithms for Clustering Data*. Englewood Cliffs, Prentice Hall.

Jarzombek, J., C.-M. Hecker, J. Wieczorek, L. Karajannis, A. Koseska, K. Ickstadt und P. Bastiaens (2014). A negative feedback in the yeast MAPK module determines where and how to become a shmoo. *nicht veröffentlichtes Manuskript, TU Dortmund & MPI Dortmund*.

Joe, H. (1997). *Multivariate Models and Dependence Concepts*. Chapman & Hall, Boca Raton.

Kalisch, M. und P. Bühlmann (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research 8*, 613–636.

Kanehisa, M. und S. Goto (2000). KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research 28*, 27–30.

Karp, P., M. Riley, S. Paley, A. Pellegrini-Toole und M. Krummenacker (1999). EcoCyc: Encyclopedia of Escherichia coli genes and metabolism. *Nucleic Acids Research 21(1)*, 55–58.

Küffner, R., T. Petri, L. Windhager und R. Zimmer (2010). Petri Nets with Fuzzy Logic (PNFL): reverse engineering and parametrization. *PLoS ONE 5*(9).

Kim, S., W. Pan und X. Shen (2013). Network-based penalized regression with application to genomic data. *Biometrics 69*(3), 582–593.

Kingman, J. F. C. (1967). Completely random measures. *Pacific Journal of Mathematics 21*, 59–78.

Kingman, J. F. C. (1978). Random discrete distributions (with discussion). *Journal of the Royal Statistical Society, Series B 37*, 1–22.

Kirk, P. D. W. und M. P. H. Stumpf (2009). Gaussian process regression bootstrapping: exploring the effects of uncertainty in time course data. *Bioinformatics 25*, 1300–1306.

Koch, I., W. Reisig und F. Schreiber (2011). *Modeling in Systems Biology: The Petri Net Approach*. Springer.

Kolpakov, F., E. Ananko, G. Kolesov und N. Kolchanov (1999). GeneNet: A gene network database and its automated visualization. *Bioinformatics 14(6)*, 529–537.

Korwar, R. M. und M. Hollander (1973). Contributions to the theory of Dirichlet processes. *Annals of Probability 1*, 705–711.

Krämer, N., J. Schäfer und A.-L. Boulesteix (2009). Regularized estimation of large-scale gene association networks using graphical Gaussian models. *BMC Bioinformatics 10*(1), 384.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady 10*(8), 707–710.

Li, N. a., A. Batzer, R. Daly, V. Yajnik, E. Skolnik, P. Chardin, D. Bar-Sagi, B. Margolis und J. Schlessinger (1993). Guanine-nucleotide-releasing factor hSos1 binds to Grb2 and links receptor tyrosine kinases to Ras signalling. *Nature 363*, 85–88.

Lippincott-Schwartz, J. (2003). Development and use of fluorescent protein markers in living cells. *Science 300*(4), 87–92.

Lohr, M., P. Godoy, J. G. Hengstler, J. Rahnenführer und M. Grzegorczyk (2010). Extracting differential regulatory sub-networks from genome-wide microrarray expression data. In *Proceedings of the Seventh International Workshop on Computational Systems Biology (WCSB 2010)*, 63–66.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, 281–297. University of California Press.

Madhamshettiwar, P., S. Maetschke, M. Davis, A. Reverter und M. Ragan (2012). Gene regulatory network inference: evaluation and application to ovarian cancer allows the prioritization of drug targets. *Genome Medicine 4(5)*.

Madigan, D. und J. York (1995). Bayesian graphical models for discrete data. *International Statistical Review 63*, 215–232.

Maeder, C. I., M. A. Hink, A. Kinkhabwala, R. Mayr, P. I. H. Bastiaens und M. Knop (2007, November). Spatial regulation of Fus3 MAP kinase activity through a reaction-diffusion mechanism in yeast pheromone signalling. *Nature Cell Biology 9*(11), 1319–1326.

Magde, D., E. Elson und W. W. Webb (1972). Thermodynamic fluctuations in a reacting system—measurement by fluorescence correlation spectroscopy. *Physical Review Letters 29*, 705–708.

Maglott, D., J. Ostell, K. Pruitt und T. Tatusova (2005). Entrez gene: gene-centered information at NCBI. *Nucleic Acids Research 33*.

Marbach, D., R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano und G. Stolovitzky (2010). Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the National Academy of Sciences 107*(14), 6286–6291.

Margolin, A. und A. Califano (2007). Theory and limitations of genetic network inference from microarray data. *Annals of the New York Academy of Sciences 1115*, 51–72.

Margolin, A., I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Favera und A. Califano (2006). ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics 7*, (Suppl 1) :S7.

Markowetz, F. und R. Spang (2007). Inferring cellular networks – a review. *BMC Bioinformatics 8*, (Suppl 6): S5.

MATLAB (2009). *Version 7.9.0.529 (R2009b)*. Natick, Massachusetts: The MathWorks Inc.

Mazur, J., D. Ritter, G. Reinelt und L. Kaderali (2009). Reconstructing nonlinear dynamic models of gene regulation using stochastic sampling. *BMC Bioinformatics 10:448*.

Meinshausen, N. und P. Bühlmann (2006). High dimensional graphs and variable selection with the Lasso. *Annals of Statistics 34*, 1436–1462.

Meyer, P., T. Cokelaer, D. Chandran, K. H. Kim, P.-R. Loh, G. Tucker, M. Lipson, B. Berger, C. Kreutz, A. Raue et al. (2014). Network topology and parameter estimation: from

experimental design methods to gene regulatory network kinetics using a community based approach. *BMC systems biology 8*(1), 13.

Needham, C., J. Bradford, A. Bulpitt und D. Westhead (2006). Inference in Bayesian networks. *Nature Biotechnology 24*(1), 51–53.

Newman, M. (2010). *Networks: An Introduction*. Oxford University Press.

Nobile, A. und A. T. Fearnside (2007). Bayesian finite mixtures with an unknown number of components. *Statistics and Computing 17*, 147–162.

Nocedal, J. und S. J. Wright (1999). *Numerical Optimization*. Springer New York.

Pearl, J. (1985). A model of self-activated memory for evidential reasoning. In *Proceedings of the 7th Conference of the Cognitive Science Society*, University of California, Irvine, CA, 329–334.

Perman, M., J. Pitman und M. Yor (1992). Size-biased sampling of Poisson point processes and excursions. *Probability and Related Fields 92*, 21–39.

Peterson, J. L. (1978). Introduction to Petri nets. *Proceedings of the National Electronics Conference 32*, 144–148.

Pitman, J. (1995). Exchangeable and partially exchangeable random partitions. *Probability Theory and Related Fields 102*(2), 145–158.

Pitman, J. (2003). Poisson-Kingman partitions. In *Darlene R. Goldstein, Hg. Statistics and science: a Festschrift for Terry Speed*, 1–34.

Qi, M. und E. A. Elion (2005). Formin-induced actin cables are required for polarized recruitment of the Ste5 scaffold and high level activation of MAPK Fus3. *J. Cell Sci. 118*, 2837–2848.

Qiu, M.-S. und S. H. Green (1992). PC12 cell neuronal differentiation is associated with prolonged p21ras activity and consequent prolonged ERK activity. *Neuron 9*(4), 705–717.

R Development Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Rabus, R., K. Trautwein und L. Wöhlbrand (2014). Towards habitat-oriented systems biology of *Aromatoleum aromaticum* EbN1: chemical sensing, catabolic network modulation and growth control in anaerobic aromatic compound degradation. *Applied Microbiology and Biotechnology*.

Ravikumar, P., M. J. Wainwright und J. D. Lafferty (2010). High-dimensional Ising model selection using ℓ_1-regularized logistic regression. *Annals of Statistics 38*(3), 1287–1319.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics 20*, 53–65.

Rozakis-Adcock, M., R. Fernley, J. Wade, T. Pawson und D. Bowtell (1993). The SH2 and SH3 domains of mammalian Grb2 couple the EGF receptor to the Ras activator mSos1. *Nature 363*, 83–85.

Ruths, D., M. Muller, J.-T. Tseng, L. Nakhleh und P. T. Ram (2008, 02). The signaling Petri net-based simulator: A non-parametric strategy for characterizing the dynamics of cell-specific signaling networks. *PLOS Computational Biology 4*(2).

Sackmann, A., M. Heiner und I. Koch (2006). Application of Petri net based analysis techniques to signal transduction pathways. *BMC Bioinformatics 7*.

Salgado, H., A. Santos, U. Garza-Ramos, J. van Helden, E. Diaz und J. Collado-Vides (2000). RegulonDB: A data base on transcription regulation in Escherichia coli. *Nucleic Acids Research 27(1)*, 59–60.

Santos, S., P. Verveer und P. Bastiaens (2007). Growth factor-induced MAPK network topology shapes Erk response determining PC-12 cell fate. *Nature Cell Biology 9*(3), 324–330.

Sasagawa, S., Y. Ozaki, K. Fujita und S. Kuroda (2005). Prediction and validation of the distinct dynamics of transient and sustained Erk activation. *Nature Cell Biology 7*, 365–373.

Schäfer, J. und K. Strimmer (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology 4 (32)*.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica 4*, 639–650.

Shachter, R. und C. Kenley (1989). Gaussian influence diagrams. *Management Science 35*, 527–550.

Shaner, N. C., G. H. Patterson und M. W. Davidson (2007). Advances in fluorescent protein technology. *Journal of Cell Science 120*(24), 4247–4260.

Shanno, D. F. (1970). Conditioning of quasi-Newton methods for function minimization. *Mathematics of Computation. 24*, 647–656.

Shannon, C. (1948). A mathematical theory of communication. *The Bell System Technical Journal 27*, 379–423.

Shotwell, M. S., K. J. Drake, V. Y. Sidorov und J. P. Wikswo (2013). Mechanistic analysis of challenge–response experiments. *Biometrics 69 (3)*, 741–747.

Shoudan, L. (1998;). Reveal, a general reverse engineering algorithm for inference of genetic network architectures. *Pacific Symposium on Biocomputing 3*, 18–29.

Silva, M. (2013). Half a century after Carl Adam Petri's Ph.D. thesis: A perspective on the field. *Annual Reviews in Control 37*(2), 191–219.

Simonson, T. und D. Perahia (1992). Normal modes of symmetric protein assemblies. application to the tobacco mosaic virus protein disk. *Biophys J. 61*(2), 410–427.

Stifanelli, P., T. Creanza, R. Anglani, V. Liuzzi, S. Mukherjee, F. Schena und N. Ancona (2013). A comparative study of covariance selection models for the inference of gene regulatory networks. *Journal of Biomedical Informatics 46*(5), 894–904.

Stlovitzky, G., D. Monroe und A. Califano (2007). Dialogue on reverse-engineering assessment and methods. *Annals of the New York Academy of Sciences 1115*(1), 1–22.

Stower, H. (2014). Metabolically constrained regulatory networks. *Nature Reviews Genetics 15 (65)*.

Tenenhaus, A., V. Guillemot, X. Gidrol und V. Frouin (2010). Gene association networks from microarray data using a regularized estimation of partial correlation based on pls regression. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on 7*(2), 251–262.

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73*(3), 273–282.

Traverse, S., N. Gomez, H. Paterson, C. Marshall und P. Cohen (1992). Sustained activation of the mitogen-activated protein (MAP) kinase cascade may be required for differentiation of PC12 cells. Comparison of the effects of nerve growth factor and epidermal growth factor. *Biochemical Journal 288*, 351–355.

Vaudry, D., P. J. S. Stork, P. Lazarovici und L. E. Eiden (2002). Signaling pathways for PC12 cell differentiation: making the right connections. *Science 296*(5573), 1648–1649.

Villaverde, A. F., J. Ross und J. R. Banga (2013). Reverse engineering cellular networks with information theoretic methods. *Cells 2*, 306–329.

Wainwright, M. J., P. Ravikumar und J. D. Lafferty (2007). High-dimensional graphical model selection using ℓ_1- regularized logistic regression. In *Advances in Neural Information Processing Systems*, 1465–1472.

Ward, J. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association 58*, 236–244.

Waterman, H. und Y. Yarden (2001). Molecular mechanisms underlying endocytosis and sorting of ErbB receptor tyrosine kinases. *FEBS letters 490*(3), 142–152.

Wentao, Z., E. Serpedin und E. R. Dougherty (2006). Inferring gene regulatory networks from time series data using the minimum description length principle. *Bioinformatics 22(17)*, 2129–2135.

Werhli, A. (2012). Comparing the reconstruction of regulatory pathways with distinct Bayesian networks inference methods. *BMC Genomics 13*(Suppl 5), S2.

Wieczorek, J., R. Malik-Sherriff, Y. Fermin, H. Grecco, E. Zamir und K. Ickstadt (2015). Uncovering distinct protein-network topologies in heterogeneous cell populations. *BMC Systems Biology 4*(9).

Wolff, F. (2013). Validierung und Fortentwicklung eines bestehenden Konzeptes zur Schätzung der Verteilung von molekularen Konzentrationen aus Fluoreszenzkorrelationsspektroskopiedaten, Technische Universität Dortmund, Bachelorarbeit.

Zhao, Y. und X. Qiao (2013). Review of modeling methods of gene expression regulation networks. *Applied Mechanics and Materials 433-435*, 783–787.

Zhou, S., S. Geer und P. Buhlmann (2009). Adaptive Lasso for high dimensional regression and Gaussian graphical modeling. arXiv:0903.2515v1.

Zou, H. (2006). The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association 101*(476), 1418–1429.

A. Algorithmus zum Clustern von Graphen

Im Folgenden wird der Algorithmus zur Berechnung paarweiser Abstände zwischen Graphen vorgestellt, vergleiche auch Abschnitt 3.4.

Definitionen	
DAG_1, DAG_2	zwei gerichtete azyklischer Graphen mit gleichen Knoten, in Form einer Adjazenzmatrix, zwischen denen der Abstand gesucht wird
Data_1	Datensatz, aus dem DAG_1 geschätzt wurde
Data_2	Datensatz, aus dem DAG_2 geschätzt wurde ($\mathrm{Data}_1 = \mathrm{Data}_2$ ist zulässig)
S, R, Q	Menge von DAGs
W	numerischer Vektor variabler Länge
M_{DAG_*}	Menge aller Nachbargraphen von DAG_*
$\mathcal{F}_1(\mathrm{DAG}_*, \mathrm{DAG}_2)$	$\sum \lvert \mathrm{DAG}_* - \mathrm{DAG}_2 \rvert$
$\mathcal{F}_2(\mathrm{DAG}_*)$	$\begin{cases} \mathcal{L}(\mathrm{DAG}_*\lvert\mathrm{Data}_1) & \text{falls } \mathrm{Data}_1 = \mathrm{Data}_2 \\ \frac{1}{2}\big(\mathcal{L}(\mathrm{DAG}_*\lvert\mathrm{Data}_1) + \mathcal{L}(\mathrm{DAG}_*\lvert\mathrm{Data}_2)\big) & \\ & \text{falls } \mathrm{Data}_1 \neq \mathrm{Data}_2 \end{cases}$ $\mathcal{L}$ ist die Likelihood eines DAGs gegeben die Daten, vergleiche Abschnitt 3, Gleichung (6).

Eingabe	
DAG_1, DAG_2, Data_1	erforderlich
Data_2	optional

Ausgabe	
Abstand	Abstand zwischen DAG_1 und DAG_2

Initiierung

$S = R = \emptyset$

$\mathrm{DAG}_1 \in Q$

$W = 0$

Berechnung

WHILE $\nexists\ \mathrm{DAG}_* \in Q : \mathcal{F}_1(\mathrm{DAG}_*, \mathrm{DAG}_2) = 0$

bestimme $M_{\mathrm{DAG}_*} \forall\ \mathrm{DAG}_* \in Q$

setze $M_{\mathrm{DAG}_*} = M_{\mathrm{DAG}_*} \backslash S$

setze $S = M_{\mathrm{DAG}_*} \cup S$

bestimme $\underset{\mathrm{DAG}_* \in M_{\mathrm{DAG}_*}}{\operatorname{argmin}} \mathcal{F}_1(\mathrm{DAG}_*, \mathrm{DAG}_2) =: N_{\mathrm{DAG}_*} \subset M_{\mathrm{DAG}_*}$

bestimme $\underset{\mathrm{DAG}_* \in N_{\mathrm{DAG}_*}}{\operatorname{argmax}} \mathcal{F}_2(\mathrm{DAG}_*, \mathrm{DAG}_2) =: R_{\mathrm{DAG}_*} \subset N_{\mathrm{DAG}_*}$

erweitere Vektor W um den Wert $\max\{\mathcal{F}_2(\mathrm{DAG}_*) : \mathrm{DAG} \in N_{\mathrm{DAG}_*}\}$

setze $Q = R_{\mathrm{DAG}_*}$

ELSE Ausgabe von Abstand := sum(W)

Die Menge der zu untersuchenden Graphen ist endlich. Somit ist die Konvergenz des Algorithmus sicher.

Definiert wird eine Abstandsfunktion *dist* ($dist : DAG \times DAG \to \mathbb{R}^+$), welche die Eigenschaften einer Semimetrik erfüllt, das heißt

$$dist(DAG_a, DAG_a) = 0\ \forall a \in \mathbb{N},$$
$$dist(DAG_a, DAG_b) = dist(DAG_b, DAG_a)\ \forall (a, b) \in \mathbb{N},$$
$$dist(DAG_a, DAG_b) \geq 0\ \forall (a, b) \in \mathbb{N}$$

und somit zur Konstruktion einer Abstandsmatrix und anschließend zur Durchführung einer Clusterung geeignet ist (Jain und Dubes, 1988).

B. Komplexeschätzer Algorithmus

Definitionen	
$\wp_1, \ldots, \wp_z, \ldots, \wp_\mathfrak{n}$	Proteine
$\mathbf{y}$	Vektor der FCS-Messungen der Länge $\sum_{z=1}^{\mathfrak{n}-1} \sum_{z^*=z+1}^{\mathfrak{n}} \mathfrak{n}_{zz^*}$, vergleiche Abschnitt 5.3
n_{boot}	Anzahl der Bootstrap-Stichproben
$\mathcal{K}_1, \ldots, \mathcal{K}_\psi, \ldots, \mathcal{K}_\Psi$	Komplexe, die aus $\wp_1, \ldots, \wp_\mathfrak{n}$ gebildet werden können ($\Psi = \sum_{z=1}^{\mathfrak{n}} \binom{\mathfrak{n}}{z}$, vergleiche Abschnitt 5.3)
$\mathcal{K}_\psi$	Konzentration des Komplexes $\mathscr{K}_\psi$
$\boldsymbol{\beta} := (\mathcal{K}_1, \ldots, \mathcal{K}_\Psi)$	Vektor der geschätzten Konzentrationen von $\mathscr{K}_1 \ldots \mathscr{K}_\Psi$
$\mathbf{X}$	Designmatrix mit $2^\mathfrak{n} - 1$ Spalten und $3 \cdot \sum_{z=1}^{\mathfrak{n}-1} \sum_{z^*=z+1}^{\mathfrak{n}} \mathfrak{n}_{zz^*}$ Zeilen.
n_{st}	Größe der Bootstrap-Stichproben
$\mathcal{B}$	Matrix $\mathrm{n}_{boot} \times \Psi$

Eingabe
$\wp_1, \ldots, \wp_\mathfrak{n}$
$\mathbf{y}$
n_{boot}
n_{st}

Ausgabe	
$\boldsymbol{\beta}$	Vektor der geschätzten Konzentrationen von $\mathscr{K}_1 \ldots \mathscr{K}_\Psi$

Initiierung	
$\mathscr{K}_1, \ldots, \mathscr{K}_\Psi$	werden aus $\wp_1, \ldots, \wp_\mathfrak{n}$ erstellt
$\mathcal{B}$	wird als Nullmatrix initiiert

Berechnung	
FOR:	$\mathfrak{j} = 1, \ldots, \mathfrak{n}_{boot}$

erstelle Bootstrap-Stichprobe $\mathbf{y}_\mathfrak{j}$ der Größe n_{st} aus $\mathbf{y}$

erstelle Designmatrix $\boldsymbol{X}_\mathfrak{j}$ kompatibel zu $\mathbf{y}_\mathfrak{j}$ gemäß Gleichung (18)

bestimme $\boldsymbol{\beta}_\mathfrak{j}$ als Lösung von $\underset{\beta_\mathfrak{j}}{\operatorname{argmin}} \|\mathbf{y}_\mathfrak{j} - \mathbf{X} \cdot \boldsymbol{\beta}_\mathfrak{j}\|_2$, vergleiche Gleichung (19)

ersetzte $\mathfrak{j}$-te Zeile von $\mathcal{B}$ durch $\boldsymbol{\beta}_\mathfrak{j}$

END

bestimme $\boldsymbol{\beta}$ als winsorisiertes Mittel der Spalten von $\mathcal{B}$

gebe $\boldsymbol{\beta}$ aus

Da der Algorithmus mit einer vorgegebenen Anzahl an Iterationen initiiert wird, ist seine Konvergenz sicher.

C. Algorithmus zur NPBN-Analyse von Netzwerkdaten

Definitionen	
$\mathcal{G}$	DAG des betrachteten Netzwerks mit d Knoten (vgl. Seite 7)
$\boldsymbol{l}$	Allokationsvektor der Länge n,
	beschreibt die Zuordnung der n Beobachtungen zu den N Komponenten
$\boldsymbol{l}_{\text{MCMC}}$	MCMC-Kette der Allokationsvektoren der Länge n_{M}
$\mathcal{G}_{\text{MCMC}}$	MCMC-Kette der DAGs der Länge n_{M}
data	Matrix der Beobachtungen/Messungen des zu untersuchenden Netzwerks
	mit n Zeilen/Beobachtungen und d Spalten/Knoten, (vgl. Seite 8)
DAG Schritt	Veränderung des DAGs durch eine Kantenoperation,
	(Entfernen, Hinzufügen, Umdrehen einer Kante vgl. Seite 8)
Allokations-	Veränderung des Allokationsvektors, es werden fünf Schritte unterschieden:
schritt	M1 Schritt, M2 Schritt, Gibbs Schritt, Verschmelzungsschritt und
	Teilungsschritt, vergleiche Abschnitt 3.2 sowie Abschnitt C (ab Seite 102)
DAG-AW	Akzeptanzwahrscheinlichkeit für einen DAG Schritt,
	vergleiche Abschnitt 3.2, Gleichung (15)
ALV-AW	Akzeptanzwahrscheinlichkeit für einen Allokationsschritt,
	vergleiche Abschnitt 3.2, Gleichung (16)
$\mathcal{P}$	Wahrscheinlichkeitsverteilung für die Ausführung der MCMC-Schritte
	$\mathcal{P} = (\mathcal{p}_1, \ldots, \mathcal{p}_5)$ mit $\sum \mathcal{p} = 1$
	$\mathcal{p}_1$: DAG Schritt, $\mathcal{p}_2$: M1 Schritt, $\mathcal{p}_3$: M2 Schritt,
	$\mathcal{p}_4$: Gibbs Schritt, $\mathcal{p}_5$: Verschmelzungsschritt/Teilungsschritt
$N_{\max}$	Anzahl der maximal zulässigen Komponenten
n_{M}	Anzahl der gewünschten MCMC-Iterationen
$n_{\text{burn-in}}$	Anzahl der Burn-In-Iterationen
n_h	Anzahl der Beobachtungen in Komponente h

Eingabe	
data	Matrix der Beobachtungen, Standardisierung wird empfohlen
$\mathcal{G}_0$	DAG zur Initiierung der MCMC-Kette, leerer Graph (Nullmatrix) oder falls Vorwissen vorhanden auch informativer DAG möglich
$\boldsymbol{l}_0$	Allokationsvektor zur Initiierung der MCMC-Kette, dies impliziert eine Festlegung von N Empfehlung: Ausgabe k-means(data) mit k=10
$\mathcal{P}_1$	Empfehlung: $\wp_1 = 0.5$, $\wp_2 = \ldots = \wp_5 = 0.125$
$\mathcal{P}_2$	Empfehlung: $\wp_1 = \wp_5 = 0.5$, $\wp_2 = \ldots = \wp_4 = 0$
$N_{\max}$	
$n_{\text{Burn-In}}$	
n_{M}	

Ausgabe	
$\boldsymbol{l}_1, \ldots, \boldsymbol{l}_{n_{\text{M}}}$	Burn-In wird entfernt
$\mathcal{G}_1, \ldots, \mathcal{G}_{n_{\text{M}}}$	Burn-In wird entfernt

Initiierung
$\mathcal{G}_I = \mathcal{G}_0$
$\boldsymbol{l}_I = \boldsymbol{l}_0$

Berechnung

FOR $\boldsymbol{I} = 1, \ldots, n_{\mathrm{M}} + n_{\text{burn-in}}$

SCHRITTAUSWAHL

überprüfe die Anzahl der Komponenten in $\boldsymbol{l_I}$ und aktualisiere N

wähle zufällig einen Schritt aus gemäß $\begin{cases} \mathcal{P}_1 & \text{, falls } N \geq 2 \\ \mathcal{P}_2 & \text{, falls } N < 2 \end{cases}$

SCHRITTAUSFÜHRUNG

IF: DAG Schritt ausgewählt

wähle unter den Nachbargraphen von $\mathcal{G}_{I-1}$ zufällig einen aus: $\mathcal{G}^v$

prüfe ob $\mathcal{G}^v$ in die MCMC-Kette gemäß DAG-AW aufgenommen wird

setze $\mathcal{G}_I := \begin{cases} \mathcal{G}_{I-1} & \text{, falls abgelehnt} \\ \mathcal{G}^v & \text{, falls akzeptiert} \end{cases}$, setze $\boldsymbol{l}_I = \boldsymbol{l}_{I-1}$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END IF

IF: M1 Schritt ausgewählt

erzeuge neuen Allokationsvektor $\boldsymbol{l}^v = \boldsymbol{l}_{I-1}$

ziehe eine Zufallszahl ξ aus Beta(1,1)

wähle in $\boldsymbol{l}^v$ zufällig zwei Komponenten h und h^* aus, $h \neq h^*$

versetze zufällig jede Beobachtung aus den Komponenten h und h^*

$\begin{cases} \text{in Komponente } h \text{ mit Wahrscheinlichkeit } \xi \\ \text{in Komponente } h^* \text{ mit Wahrscheinlichkeit } 1 - \xi \end{cases}$

prüfe ob $\boldsymbol{l}^v$ in die MCMC-Kette gemäß ALV-AW aufgenommen wird

die zugehörige Vorschlagswahrscheinlichkeit lautet

$\frac{q(\boldsymbol{l}^v|\boldsymbol{l}_{I-1})}{q(\boldsymbol{l}_{I-1}|\boldsymbol{l}^v)} = \frac{\Gamma(\theta+n_h)\Gamma(\theta+n_{h^*})}{\Gamma(\theta+\tilde{n}_h)\Gamma(\theta+\tilde{n}_{h^*})}$, $\tilde{n}_h$ und $\tilde{n}_{h^*}$ geben die Anzahl der Beobachtungen in den neuallokierten Komponenten an

Berechnung, Fortsetzung

setze $\boldsymbol{l}_I := \begin{cases} \boldsymbol{l}_{I-1}\ , & \text{falls abgelehnt} \\ \boldsymbol{l}^v\ , & \text{falls akzeptiert} \end{cases}$, setze $\mathcal{G}_I = \mathcal{G}_{I-1}$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END IF

IF: M2 Schritt ausgewählt

erzeuge neuen Allokationsvektor $\boldsymbol{l}^v = \boldsymbol{l}_{I-1}$

wähle aus $\boldsymbol{l}^v$ zufällig zwei Komponenten h und h^* aus, $h \neq h^*$

versetze zufällig u Beobachtungen aus Komponente h in Komponente h^*

u folgt der Gleichverteilung auf $\{1, \ldots, n_h\}$

prüfe ob $\boldsymbol{l}^v$ in die MCMC-Kette gemäß ALV-AW aufgenommen wird

die zugehörige Vorschlagswahrscheinlichkeit lautet

$$\frac{q(\boldsymbol{l}^v|\boldsymbol{l}_{I-1})}{q(\boldsymbol{l}_{I-1}|\boldsymbol{l}^v)} = \frac{n_h}{n_{h^*}+u} \cdot \frac{n_h! n_{h^*}!}{(n_h-u)!(n_{h^*}+u)!}$$

setze $\boldsymbol{l}_I := \begin{cases} \boldsymbol{l}_{I-1}\ , & \text{falls abgelehnt} \\ \boldsymbol{l}^v\ , & \text{falls akzeptiert} \end{cases}$, setze $\mathcal{G}_I = \mathcal{G}_{I-1}$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END IF

IF: Gibbs Schritt ausgewählt

wähle zufällig eine Beobachtung $\mathbf{d}$ aus data aus

erzeuge neue Allokationsvektoren $\boldsymbol{l}^h = \boldsymbol{l}_{I-1}$ für $h = 1, \ldots, N$

versetze $\mathbf{d}$ in jedem der $\boldsymbol{l}^h$ in die Komponente h

bestimme $\mathcal{F}_{\boldsymbol{l}}$, die zugehörige vollständig bedingte Verteilung der $\boldsymbol{l}^h$

ziehe $\boldsymbol{l}_I$ aus $\mathcal{F}_{\boldsymbol{l}}$, setze $\mathcal{G}_I = \mathcal{G}_{I-1}$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END IF

Berechnung, Fortsetzung

IF: Teilungs- oder Verschmelzungsschritt ausgewählt

wähle zufällig mit Wahrscheinlichkeit p_N^{Teilung} den Teilungsschritt

oder mit Wahrscheinlichkeit $p_N^{\text{Verschmelzung}}$ den Verschmelzungsschritt aus

dabei ist $p_N^{\text{Teilung}} = \begin{cases} 0.5 & N = 2, \ldots, N_{\max} - 1 \\ 1 & N = 1 \\ 0 & N = N_{\max} \end{cases}$ und $p_N^{\text{Verschmelzung}} = 1 - p_N^{\text{Teilung}}$

IF: Verschmelzungsschritt ausgewählt

erzeuge neuen Allokationsvektor $\boldsymbol{l}^v = \boldsymbol{l}_{I-1}$

wähle in $\boldsymbol{l}^v$ zufällig zwei Komponenten h und h^* aus, $h \neq h^*$

versetzte alle Beobachtungen aus Komponente h^* in Komponente h und lösche h^*

prüfe ob $\boldsymbol{l}^v$ in die MCMC-Kette gemäß ALV-AW aufgenommen wird

die zugehörige Vorschlagswahrscheinlichkeit lautet:

$$\frac{q(\boldsymbol{l}^v|\boldsymbol{l}_{I-1})}{q(\boldsymbol{l}_{I-1}|\boldsymbol{l}^v)} = \frac{p_N^{\text{Verschmelzung}}}{1-p_N^{\text{Verschmelzung}}} \frac{\Gamma(2\theta)}{\Gamma(\theta)\Gamma(\theta)} \frac{\Gamma(2\theta+n_h)}{\Gamma(\theta+\tilde{n}_h)\Gamma(\theta+\tilde{n}_{h^*})}$$

setze $\boldsymbol{l}_I := \begin{cases} \boldsymbol{l}_{I-1}\,, & \text{falls abgelehnt} \\ \boldsymbol{l}^v\,, & \text{falls akzeptiert} \end{cases}$, setze $\mathcal{G}_I = \mathcal{G}_{I-1}$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END

Berechnung, Fortsetzung

IF: Teilungsschritt ausgewählt

erzeuge neuen Allokationsvektor $\boldsymbol{l}^v = \boldsymbol{l}_{I-1}$

wähle in $\boldsymbol{l}^v$ zufällig eine Komponente h aus

ziehe eine Zufallszahl ξ aus Beta(1,1)

erzeuge eine neue (leere) Komponente h^*

versetze zufällig jede Beobachtungen aus der Komponente h

$$\begin{cases} \text{in Komponente } h\ , & \text{mit Wahrscheinlichkeit } \xi \\ \text{in Komponente } h^*\ , & \text{mit Wahrscheinlichkeit } 1-\xi \end{cases}$$

permutiere zufällig die Benennung aller Komponenten

prüfe ob $\boldsymbol{l}^v$ in die MCMC-Kette gemäß ALV-AW aufgenommen wird

die zugehörige Vorschlagswahrscheinlichkeit lautet:

$$\frac{q(\boldsymbol{l}^v|\boldsymbol{l}_{I-1})}{q(\boldsymbol{l}_{I-1}|\boldsymbol{l}^v)} = \frac{1-p_N^{\text{Teilung}}}{p_N^{\text{Teilung}}} \frac{\Gamma(\theta)\Gamma(\theta)}{\Gamma(2\theta)} \frac{\Gamma(2\theta+n_h)}{\Gamma(\theta+\tilde{n}_h)\Gamma(\theta+\tilde{n}_{h^*})}$$

$$\text{setze } \boldsymbol{l}_I := \begin{cases} \boldsymbol{l}_{I-1}\ , & \text{falls abgelehnt} \\ \boldsymbol{l}^v\ , & \text{falls akzeptiert} \end{cases}, \quad \text{setze } \mathcal{G}_I = \mathcal{G}_{I-1}$$

setze $I = I + 1$, GOTO SCHRITTAUSWAHL

END

END IF

END

lösche die Einträge $\boldsymbol{l}_0, \ldots, \boldsymbol{l}_{n_{\text{burn-in}}}$ aus der MCMC-Kette der Allokationsvektoren

sowie $\mathcal{G}_0, \ldots, \mathcal{G}_{n_{\text{burn-in}}}$ aus der MCMC-Kette der DAGs

gebe $\boldsymbol{l}_{\text{MCMC}}$ sowie $\mathcal{G}_{\text{MCMC}}$ aus

Da der Algorithmus mit einer vorgegebenen Anzahl an Iterationen initiiert wird, ist seine Konvergenz sicher.

D. Ergänzende Details für das Erk-Signalübertragungsnetzwerk-Modell

In seiner Arbeit führt Sasagawa et al. (2005) die bis dahin bekannten Modelle von Teilen des Erk-Signalübertragungsnetzwerks zusammen und erstellt daraus ein alle Spezies umfassendes Gesamtmodell. Aus den Vorarbeiten war die Interaktionsstruktur der Spezies bis auf wenige Ausnahmen, welche bei der Ausarbeitung des Modells experimentell geklärt werden konnten, bekannt (Vaudry et al., 2002; Gotoh et al., 1990; Qiu und Green, 1992; Traverse et al., 1992; Egan et al., 1993; Rozakis-Adcock et al., 1993; Li et al., 1993; Gale et al., 1993; Davis, 1993; Di Fiore und Gillt, 1999; Waterman und Yarden, 2001).
Bei der Erstellung des Gesamtmodells bestand eine der Hauptschwierigkeiten darin, die mehrheitlich parallel ablaufenden Reaktionen aufeinander abzustimmen und die Geschwindigkeit, in der sie relativ zueinander ablaufen, die sogenannte Kinetik, kohärent zu bestimmen. Zur Lösung dieses Problems wurde von Sasagawa auf Differentialgleichungen zurückgegriffen, von denen die wichtigsten hier vorgestellt werden sollen.

Die folgenden drei Gleichungen beschreiben die *Ras* und *Rap*1 Aktivierung. Diese wird durch die $pR-$abhängige Aktivierung von GEF und GAP reguliert. Die Derivate (durch chemische Reaktionen abgewandelte Stoffe) von $pR-GEF$, $pR-GAP$ und $GTPase-GTP$ sind gegeben durch

$$\frac{d[pR-GEF]}{dt} = k_1[pR]([GEF_{Total}]-[pR-GEF]) - k_2[pR-GEF] \ ,$$

$$\frac{d[pR-GAP]}{dt} = k_3[pR]([GAP_{Total}]-[pR-GAP]) - k_4[pR-GAP] \ ,$$

$$\begin{aligned}\frac{d[GTPase-GTP]}{dt} &= k_5[pR-GEF]([GTPase_{Total}]-[GTPase-GTP]) \\ &\quad - k_6[pR-GAP][GTPase-GTP] \ ,\end{aligned}$$

wobei [*] die Konzentration der Spezies * zum Zeitpunkt t beschreibt. Die Gesamtkonzentrationen von GEF, GAP und $GTPase$ ($[GEF_{Total}]$, $[GAP_{Total}]$ und $[GTPase_{Total}]$) werden durch die Reaktion nicht verändert, ihre Verhältnisse bleiben somit durchgehend

gewahrt. In der dritten Gleichung wird implizit angenommen, dass die Konzentration der $GTPase$, gebunden entweder an GEF oder GAP, verhältnismäßig gering ist verglichen mit der Gesamtkonzentration der $GTPase$, GEF und GAP.
Die drei Gleichungen lassen sich in eine dimensionslose Form überführen:

$$\frac{dGEF}{dt} = k_2\{pR - (1 + pR)GEF\},$$

$$\frac{dGAP}{dt} = k_4\{p \times pR) - (1 + (p \times pR))GAP\},$$

$$\frac{dGTPase}{dt} = k_6[GAP_{Total}]\{GEF/Ke - (GEF/Ke + GAP)GTPase\},$$

wobei

$$\begin{aligned} pR &= [pR]/Kd \ , \\ GEF &= [pR - GEF]/[GEF_{Total}] \ , \\ GAP &= [pR - GAP]/[GAP_{Total}] \ , \\ GTPase(Ras \text{ oder } Rap1) &= [GTPase - GTP]/GTPase_{Total}] \ , \\ Kd &= k_2/k_1 = pk_4/k_3 \ , \\ Ke &= k_6[GAP_{Total}]/k_5[GEF_{Total}] \ . \end{aligned}$$

Der Quotient $k_6[GAP_{Total}]/k_2$ repräsentiert die Relaxations-Zeitkonstante der $GTPase$ relativ zu der von GEF, wobei GEF und GAP als konstant angenommen werden. Der Quotient k_4/k_2 kann interpretiert werden als die Reaktions-Geschwindigkeitskonstante der GAP-Aktivierung verglichen mit der Reaktions-Geschwindigkeitskonstante der GEF-Aktivierung unter der Bedingung, dass pR konstant ist und GEF und GAP ihre Sättigung nicht erreicht haben.

E. Zusammenfassung der Bachelorarbeit von Wolff (2013) zur Beurteilung der Leistungsfähigkeit des Komplexeschätzers

In diesem Abschnitt wird auf die Validierung des Komplexeschätzers eingegangen, welcher im Rahmen einer am Lehrstuhl für mathematische Statistik und biometrische Anwendungen der TU Dortmund betreuten Bachelorarbeit (Wolff, 2013) vorgenommen worden ist.

Das Ziel der Bachelorarbeit ist es die Leistungsfähigkeit der bereits vorliegenden Implementierung des Kompexeschätzers zu bewerten. Weiter sollen für die konfigurierbaren Teile des Verfahrens, wie etwa für die Kostenfunktion oder für den Optimierungsalgorithmus, vergleiche Abschnitt 5.3 ab Seite 66, im Sinne der Leistungsfähigkeit, optimale Parametereinstellungen gefunden werden. Dies beinhaltet auch die Konzeption und Durchführung einer Simulationsstudie.

Die Idee zur Generierung der Daten wurde stark von den Besonderheiten des Messverfahrens beeinflusst, welches im Abschnitt 5.3 beschrieben wird. Ferner wird die dort eingeführte Notation verwendet. Formuliert wird die Beschreibung für den allgemeinen Fall, dass Fluoreszenzkorrelationsspektroskopiemessungen (FCS-Messungen) eines biochemischen Systems simuliert werden sollen, welches aus $\mathfrak{n}$ Proteinen $(\wp_1, \ldots, \wp_z, \ldots, \wp_\mathfrak{n})$ besteht. Diese können untereinander Bindungen eingehen und so $\Psi \ \left(= \sum_{z=1}^{\mathfrak{n}} \binom{\mathfrak{n}}{z}\right)$ unterscheidbare Komplexe $\mathscr{K}_\psi$ $(\psi = 1, \ldots, \Psi)$ bilden. Die Konzentrationen dieser Komplexe wird mit $\mathcal{K}_\psi$ notiert. Diese ist jedoch nicht direkt messbar. Das FCS Verfahren kann nur die gemeinsame Konzentration so genannter Gruppen von Komplexen (y_{zz^*}) erheben (vergleiche Abschnitt 5.2 und 5.3, insbesondere Gleichung 17 auf Seite 65). Die Zusammensetzung der Gruppen hängt von den jeweils markierten Proteinen $(\wp_z, \wp_{z^*})$ ab, vergleiche auch Abbildung 17 auf Seite 116. Um ein System zu erfassen, welches von $\mathfrak{n}$ Proteinen gebildet wird, werden $\binom{\mathfrak{n}}{2}$ FSC-Messungen benötigt.

Gegeben sei ein System mit $\mathfrak{n}$ Proteinen. Von jedem möglichen Proteinpaar $\wp_z\wp_{z^*}$ sollen $\mathfrak{n}_{zz^*}$ Werte generiert werden. Der Parameter $\mathscr{N}$ steht für Noise und gibt die Stärke der

Störeinflüsse an, welche sowohl biologische als auch technische Ursachen haben können. Zuerst wird für jeden der betrachteten Komplexe $\mathscr{K}_\psi$, $\psi = 1, \ldots, \Psi$, die „wahre“ Konzentration $\widetilde{\mathcal{K}_\psi}$ aus einer Gleichverteilung gezogen $\widetilde{\mathcal{K}_\psi} \sim \mathrm{Re}[0, 100]$. Diese soll später als Vergleichswert zur Bestimmung der Qualität der Schätzung herangezogen werden.

Anschließend wird, gemäß der Vorschrift aus Gleichung 17, der simulierte Gruppenwert gebildet,

$$\tilde{y}_{zz^*} = \begin{pmatrix} \tilde{y}_{zz^*}^{AC(\wp_z)} \\ \tilde{y}_{zz^*}^{AC(\wp_{z^*})} \\ \tilde{y}_{zz^*}^{CC(\wp_z \wp_{z^*})} \end{pmatrix} = \begin{pmatrix} \sum_{\psi \in \Lambda} \widetilde{\mathcal{K}_\psi} \\ \sum_{\psi \in \Delta} \widetilde{\mathcal{K}_\psi} \\ \sum_{\psi \in \Xi} \widetilde{\mathcal{K}_\psi} \end{pmatrix},$$

wobei $\Lambda := \{\widetilde{\mathcal{K}_\psi} : \wp_z \lhd \mathscr{K}_\psi\}$, $\Delta := \{\widetilde{\mathcal{K}_\psi} : \wp_{z^*} \lhd \mathscr{K}_\psi\}$, $\Xi := \{\widetilde{\mathcal{K}_\psi} : \wp_z \lhd \mathscr{K}_\psi \wedge \wp_{z^*} \lhd \mathscr{K}_\psi\}$.

Danach wird für jede der $\binom{\mathfrak{n}}{2}$ Proteinkombinationen der simulierte Messwert gebildet, indem auf die simulierte Gruppenkonzentration noch ein normalverteiltes Rauschen hinzu addiert wird, welches die Störeinflüsse repräsentiert,

$$\check{y}_{zz^*} = \begin{pmatrix} \check{y}_{zz^*}^{AC(\wp_z)} \\ \check{y}_{zz^*}^{AC(\wp_{z^*})} \\ \check{y}_{zz^*}^{CC(\wp_z \wp_{z^*})} \end{pmatrix} = \begin{pmatrix} \tilde{y}_{zz^*}^{AC(\wp_z)} + \epsilon \ , & \text{mit } \epsilon \sim \mathrm{N}\left(0, \tilde{y}_{zz^*}^{AC(\wp_z)} \cdot \mathscr{N}\right) \\ \tilde{y}_{zz^*}^{AC(\wp_{z^*})} + \epsilon \ , & \text{mit } \epsilon \sim \mathrm{N}\left(0, \tilde{y}_{zz^*}^{AC(\wp_{z^*})} \cdot \mathscr{N}\right) \\ \tilde{y}_{zz^*}^{CC(\wp_z \wp_{z^*})} + \epsilon \ , & \text{mit } \epsilon \sim \mathrm{N}\left(0, \tilde{y}_{zz^*}^{AC(\wp_z \wp_{z^*})} \cdot \mathscr{N}\right) \end{pmatrix} .$$

Der letzte Schritt wird wiederholt, bis die gewünschte Anzahl an Beobachtungen ($\mathfrak{n}_{zz^*}$) der jeweiligen Kombination zz^* vorliegt. Der resultierende Datensatz ist ein Vektor $\mathbf{y} := \left(\check{y}_{1\ 2}^{(1)}, \check{y}_{1\ 2}^{(2)}, \ldots, \check{y}_{1\ 2}^{(\mathfrak{n}_1\ 2)}, \ldots, \check{y}_{\mathfrak{n}-1\ \mathfrak{n}}^{(1)}, \ldots, \check{y}_{\mathfrak{n}-1\ \mathfrak{n}}^{(\mathfrak{n}_{\mathfrak{n}-1}\ \mathfrak{n})}\right)^T$ mit der Länge $\sum_{z=1\ z^*=2}^{\mathfrak{n}} \mathfrak{n}_{zz^*}$, wobei $z \leq z^*$.

Im Rahmen der Bachelorarbeit werden mehrere Datensätze generiert. Betrachtet werden sowohl solche mit $\mathfrak{n} = 4$ (15 mögliche Komplexe, genauso wie im mating pathway der Hefe aus Abschnitt 6) als auch solche mit $\mathfrak{n} = 6$ (63 mögliche Komplexe). Durch Variation des Parameters $\mathscr{N} = 0.01, 0.1, 0.2, \ldots, 0.5$ können Daten unterschiedlicher Qualität simuliert werden, was eine umfassendere Bewertung der Leistungsfähigkeit des Komplexeschätzers erlaubt.

Die Bewertung erfolgt, indem der Komplexeschätzer auf die simulierten Datensätze angewendet wird und die geschätzten Konzentrationen $\widehat{\mathcal{K}_\psi}$ mit den simulierten „wahren“ Konzentrationen $\widetilde{\mathcal{K}_\psi}$ verglichen werden. Betrachtet wird sowohl der absolute Fehler $\sum_\psi^\Psi |\widetilde{\mathcal{K}_\psi} - \widehat{\mathcal{K}_\psi}|$ als auch der relative Fehler $\sum_\psi^\Psi |\widetilde{\mathcal{K}_\psi} - \widehat{\mathcal{K}_\psi}| \cdot \widetilde{\mathcal{K}_\psi}^{-1}$ der Schätzung.

Im Folgenden werden die Ergebnisse der Bachelorarbeit kurz vorgestellt. Der Vergleich der Optimierungsalgorithmen zeigt, dass Verfahren, welche keine Beschränkung des Parameterbereiches erlauben, jenen, die diese Möglichkeit bieten, deutlich unterlegen sind. Schlechte Ergebnisse, also eine Schätzung mit hohem absoluten und relativen Fehler, zeigen die Verfahren newuoa, uobyqa, SANN und Nelder-Mead. Gute Ergebnisse zeigen die Verfahren bobyqa, BFGS , L-BFGS-B und BBoptim, wobei die beiden letzteren mit einer durchschnittlichen Abweichung von unter fünf Prozent am besten abschneiden. Der Abstand zwischen den Gruppen ist recht groß. So ist der relative Fehler der nicht beschränkten Verfahren mitunter zehn mal so groß wie der der Verfahren, welche Nebenbedingungen zulassen.

Ein weiterer konfigurierbarer Parameter des Komplexeschätzers, der untersucht wird, ist der Normparameter p der Verlustfunktion ($\underset{\boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{X} \cdot \boldsymbol{\beta}\|_p, p \in \mathbb{N}$), vergleiche Gleichung 19. Im Rahmen der Bachelorarbeit werden als Normen,

$$\|x\|_p := \left(\sum_i^n |x_i|^p \right)^{\frac{1}{p}},$$

die Absolutnorm ($p = 1$) und die Euklidische Norm ($p = 2$) untersucht. Der Komplexeschätzer liefert bei Verwendung der Zweiten bessere Ergebnisse. Mitunter wird eine Verbesserung um den Faktor 2 beobachtet.

Zusätzlich zu den beschriebenen Vergleichen, bei denen ein sehr geringes Rauschen von $\mathscr{N} = 0.01$ vorliegt, da hier der Schwerpunkt auf die Optimierung der Methode gelegt wird, werden auch Szenarios mit starkem Rauschen untersucht. Für die als leistungsstark identifizierte Kombination von Verlustfunktion und Optimierungsverfahren Euklidischer Abstand und L-BFGS-B werden fünf Einstellungen von $\mathscr{N}$ betrachtet (0.1, 0.2, 0.3, 0.4, 0.5). Dabei repräsentiert 0.1 eine gute Datenqualität und 0.5 steht für stark verzerrte Messungen. Die durchgeführten Untersuchungen zeigen, dass der Komplexeschätzer auch mit starkem Rauschen umgehen kann. Zwar wird der Abstand der geschätzten Werte

zu den „wahren“ Werten mit steigendem $\mathscr{N}$ tendenziell größer, bei $\mathscr{N} = 0.5$ liegt er im Durchschnitt bei dem ca. 1.5-fachen des Fehlers, welcher bei einer Einstellung von $\mathscr{N} = 0.1$ im Durchschnitt beobachtet wird. Dennoch sind die beobachteten relativen Fehler im Durchschnitt unter zehn Prozent.

F. Abbildungen und Tabellen

Datensatz	# Iterationen	Ausdünnung	Burn-In
D2 (Zeitpunkte $1, \ldots, 10$)	$2.8 \cdot 10^6$	350	$1.4 \cdot 10^6$
D4 (Zeitpunkte $1, \ldots, 10$)	$5 \cdot 10^6$	500	$2 \cdot 10^6$
Hefe (US, SS, LS)	$3.2 \cdot 10^6$	350	$2 \cdot 10^6$
Hefe (vermischt)	$3.2 \cdot 10^6$	500	$2 \cdot 10^6$

Tabelle 4: *Übersicht der Parametereinstellungen der MCMC-Simulationen für den Erk-Signalübertragungsnetzwerk-Datensatz, mit zwei Komponenten D2 und mit vier Komponenten D4. Jeweils für die Zeitpunkte erste bist zehnte Minute nach Stimulation mit EGF bzw. NGF (vergleiche Abschnitt 4.3) sowie für den Datensatz des mating pathways in der Hefe, jeweils für nicht (US), kurz (SS) und lang (LS) stimulierte Hefen (6.2) und für den künstlich vermischten Datensatz der Hefe (6.3).*

Spezies	Ausgangskonzentration in μM	Spezies	Ausgangskonzentration in μM
EGFR	0.3	proteasome	0.0
L_EGFR	0.0	Grb2_SOS_pShc	0.0
L_EGFR_dimer	0.0	Shc_dpEGFR_c_Cbl	0.0
L_dpEGFR	0.0	Grb2_SOS_pShc_dpEGFR	0.0
NGFR	0.061894	pFRS2	0.0
pTrkA	0.0	FRS2_dpEGFR	0.0
L_NGFR	0.0	pDok_RasGAP	0.0
SOS	0.1	pFRS2_dpEGFR_c_Cbl_ubiq	0.0
pSOS	0.0	Ras_GTP	0.0
SOS_Grb2	0.0	Crk_C3G_pFRS2_dpEGFR_c_Cbl_ubiq	0.0
Grb2	1.0	c_Raf_Ras_GTP	0.0
Dok	0.3	B_Raf_Ras_GTP	0.0
pDok	0.0	ppMEK	0.0
Crk	1.0	ppERK	0.0
FRS2	1.0	Crk_C3G	0.0
Shc	1.0	Rap1_GTP	0.0
pSOS_Grb2	0.0	ppMEK_ERK	0.0
Rap1_GDP	0.2	dppERK	0.0
MEK	0.68	Shc_pTrkA	0.0
MKP3	0.018	Shc_pTrkA_endo	0.0
pShc_dpEGFR	0.0	pShc_pTrkA	0.0
dpEGFR_c_Cbl	0.0	pFRS2_pTrkA	0.0
B_Raf_Rap1_GTP	0.0	FRS2_pTrkA	0.0
pShc_dpEGFR_c_Cbl	0.0	pShc_pTrkA_endo	0.0
pFRS2_dpEGFR_c_Cbl	0.0	FRS2_pTrkA_endo	0.0
Shc_dpEGFR	0.0	pFRS2_pTrkA_endo	0.0
c_Cbl	0.5	Crk_C3G_pFRS2_pTrkA_endo	0.0
RasGAP	0.1	Grb2_SOS_pShc_pTrkA	0.0
c_Raf	0.5	Crk_C3G_pFRS2_pTrkA	0.0
B_Raf	0.2	Grb2_SOS_pShc_pTrkA_endo	0.0
ERK	0.26	c_Raf_Ras_GTP_MEK	0.0
PP2A	0.24	c_Raf_Ras_GTP_pMEK	0.0

Diese Tabelle wird auf der folgenden Seite fortgesetzt.

Spezies	Ausgangskonzentration in μM	Spezies	Ausgangskonzentration in μM
Ras_GDP	0.1	c_Raf_Ras_GTP_MEK_ERK	0.0
Rap1GAP	0.012	c_Raf_Ras_GTP_pMEK_ERK	0.0
C3G	0.5	B_Raf_Ras_GTP_MEK	0.0
pShc	0.0	B_Raf_Ras_GTP_pMEK	0.0
pFRS2_dpEGFR	0.0	B_Raf_Ras_GTP_MEK_ERK	0.0
pTrkA_endo	0.0	B_Raf_Ras_GTP_pMEK_ERK	0.0
MEK_ERK	0.0	B_Raf_Rap1_GTP_MEK	0.0
pMEK_ERK	0.0	B_Raf_Rap1_GTP_pMEK	0.0
FRS2_dpEGFR_c_Cbl_ubiq	0.0	B_Raf_Rap1_GTP_MEK_ERK	0.0
Crk_C3G_pFRS2_dpEGFR_c_Cbl	0.0	B_Raf_Rap1_GTP_pMEK_ERK	0.0
pShc_dpEGFR_c_Cbl_ubiq	0.0	ppERK_MKP3	0.0
Crk_C3G_pFRS2_dpEGFR	0.0	dppERK_MKP3	0.0
Grb2_SOS_pShc_dpEGFR_c_Cbl_ubiq	0.0	pro_TrkA	0.020631
Grb2_SOS_pShc_dpEGFR_c_Cbl	0.0	pro_EGFR	0.3
Shc_dpEGFR_c_Cbl_ubiq	0.0	degradation	0.0
dpEGFR_c_Cbl_ubiq	0.0	FRS2_dpEGFR_c_Cbl	0.0
pMEK	0.0		
EGF	0.001613 oder 0.0, abhängig von Stimulationsziel		
NGF	0.001613 oder 0.0, abhängig von Stimulationsziel		

Tabelle 5: *Übersicht aller im Erk-Signalübertragungsnetzwerk-Modell vorkommenden Spezies und deren Ausgangskonzentrationen. Sie gelten sowohl für die Zellen des Wildtyps als auch für mutierte Zellen. Eine weiterführende Beschreibung der Reaktionswege sowie der beteiligten Stoffe ist Wieczorek et al. (2015) oder Sasagawa et al. (2005) zu entnehmen. Das vollständige Modell kann unter www.ebi.ac.uk, unter BIOMD0000000049 zusammen mit Zusatzinformationen abgerufen werden.*

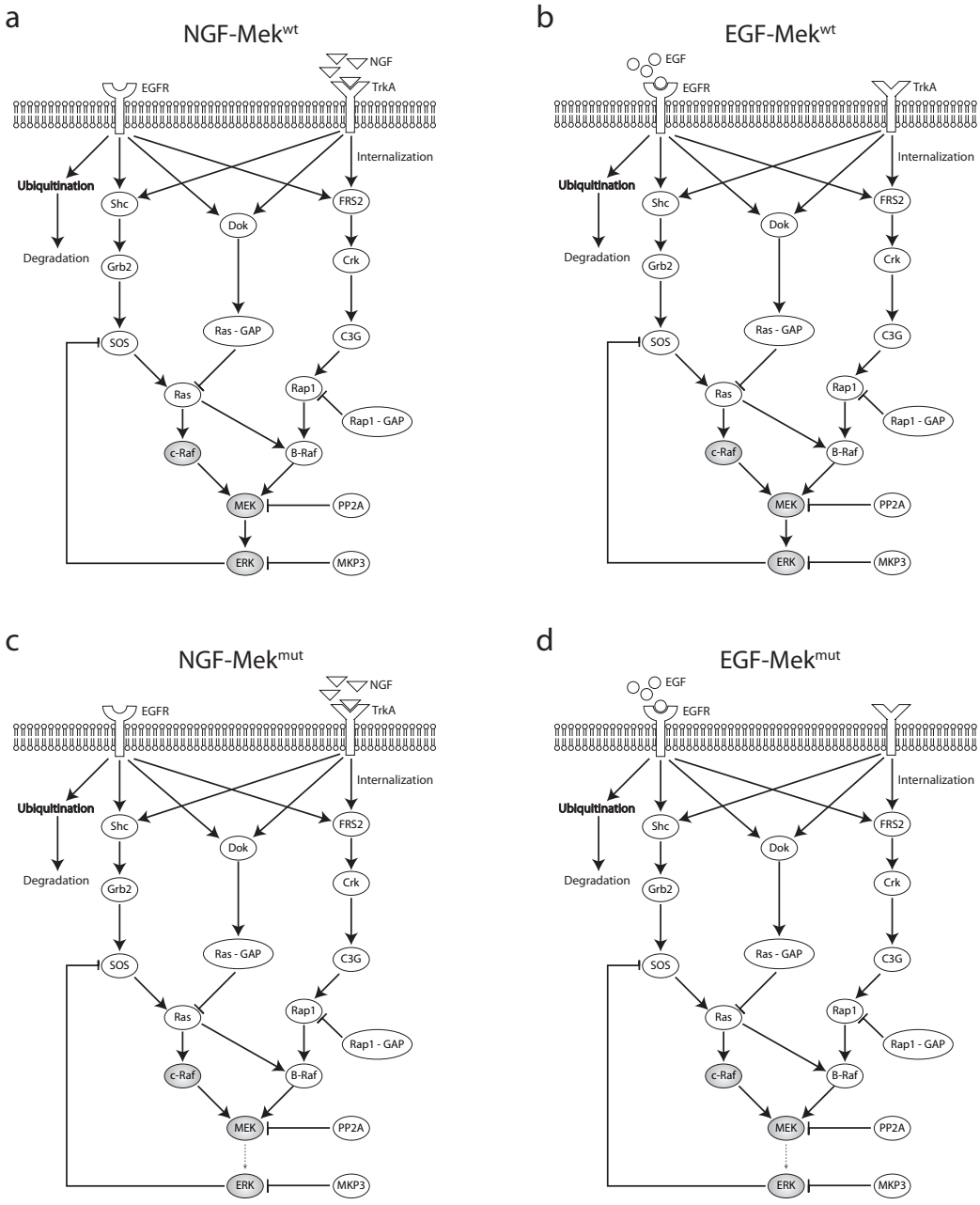


Abbildung 16: *Schematische Darstellung des simulierten Erk-Signalübertragungsnetzwerkes, Wildtyp unter Stimulation mit NGF (a), Wildtyp unter Stimulation mit EGF (b), Mutante unter Stimulation mit NGF (c), Mutante unter Stimulation mit EGF (d). Abgebildet ist ein Ausschnitt aus dem Netzwerk, welcher, stark vereinfacht, die Reaktion der Zelle des Wildtyps (a, b) und der Mutante (c, d) auf die chemischen Botenstoffe EGF (b bzw. d) und NGF (a bzw. c) darstellt. Die für diese Arbeit zentralen Spezies Raf (c-Raf), MEK, und ERK sind grau unterlegt. Quelle: Wieczorek et al. (2015).*

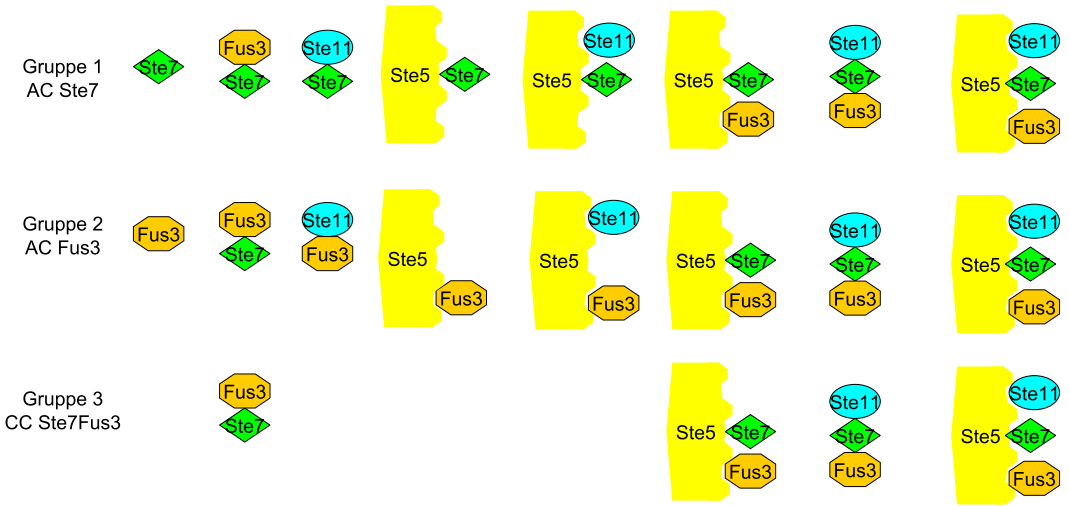


Abbildung 17: *Übersicht der Gruppen der gemessenen Konstellationen für ein Experiment mit markierten Ste7 und Fus3. Gruppe 1 korrespondiert zur Messung der Lichtwellenlänge, auf welche die Markierung, mit der Ste7 versehen wurde, reagiert. Zum Messwert von AC Ste7 tragen die Konzentrationen aller in Gruppe 1 befindlichen Proteine bei ({Ste7}, {Fus3 Ste7}, {Ste11 Ste7},{Ste5 Ste7},{Ste11 Ste7 Ste5}, {Ste5 Ste7 Fus3}, {Ste11 Ste7 Fus3}, {Ste5 Ste7 Fus3 Ste11}). Analog tragen zum Messwert von AC Fus3 die Konzentrationen aller in der Gruppe 2 befindlichen Proteine bei ({Fus3}, {Fus3 Ste7}, {Ste11 Fus3},{Ste5 Fus3},{Ste11 Fus3 Ste5}, {Ste5 Ste7 Fus3}, {Ste11 Ste7 Fus3}, {Ste5 Ste7 Fus3 Ste11}). Zum Messwert von CC Ste7 Fus3 tragen die Konzentrationen aller im Schnitt der Gruppen 1 und 2 befindlichen Proteine bei ({Fus3 Ste7}, {Ste7 Fus3 Ste5}, {Ste11 Ste7 Fus3}, {Ste5 Ste7 Fus3 Ste11}). Die exakte Konzentration von Beispielsweise {Ste7}, {Ste7 Ste11} ist aus den puren Messwerten (AC Ste7, AC Fus3 und CC Ste7 Fus3) nicht direkt feststellbar. In der Schreibweise aus Gleichung 17 hat $y_{Ste7\ Fus3}$ folgende Gestalt:*

$$\begin{pmatrix} y^{AC(\text{Ste7})}_{\text{Ste7 Fus3}} \\ y^{AC(Fus3)}_{\text{Ste7 Fus3}} \\ y^{CC(\text{Ste7 Fus3})}_{\text{Ste7 Fus3}} \end{pmatrix} = \begin{pmatrix} \mathcal{K}_4 + \mathcal{K}_9 + \mathcal{K}_7 + \mathcal{K}_{10} + \mathcal{K}_{13} + \mathcal{K}_{14} + \mathcal{K}_{12} + \mathcal{K}_{15} + \varepsilon_{\text{Ste7}} \\ \mathcal{K}_2 + \mathcal{K}_9 + \mathcal{K}_5 + \mathcal{K}_8 + \mathcal{K}_{11} + \mathcal{K}_{14} + \mathcal{K}_{12} + \mathcal{K}_{15} + \varepsilon_{\text{Fus3}} \\ \mathcal{K}_9 + \mathcal{K}_{14} + \mathcal{K}_{12} + \mathcal{K}_{15} + \varepsilon_{\text{Ste7Fus3}} \end{pmatrix} .$$

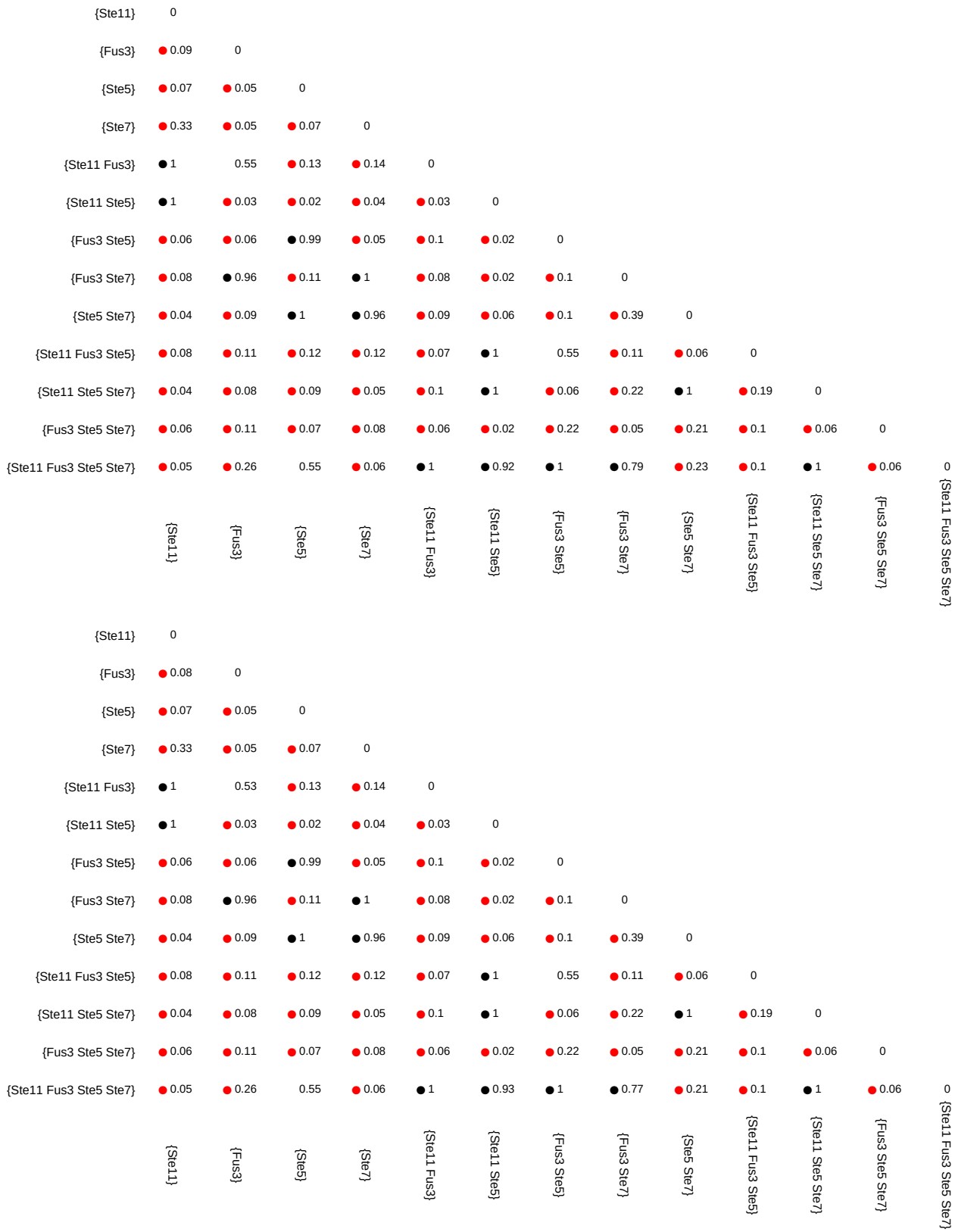


Abbildung 18: *Geschätzte pep-Matrix für die nicht stimulierte Hefe, berechnet mit NPBN-DP (oben) und NPBN-PY(unten).*

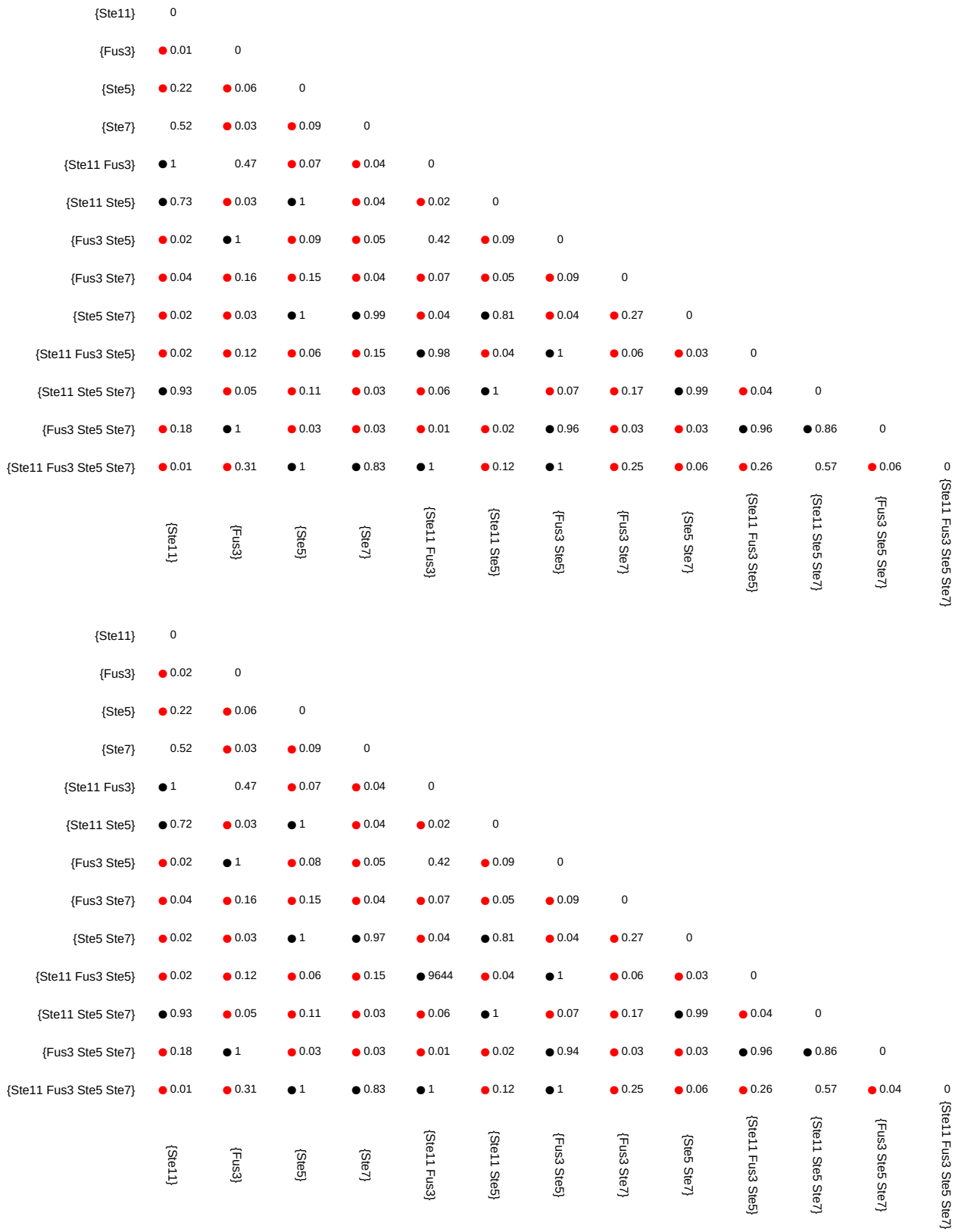


Abbildung 19: *Geschätzte pep-Matrix für die kurz stimulierte Hefe, berechnet mit NPBN-DP (oben) und NPBN-PY(unten).*

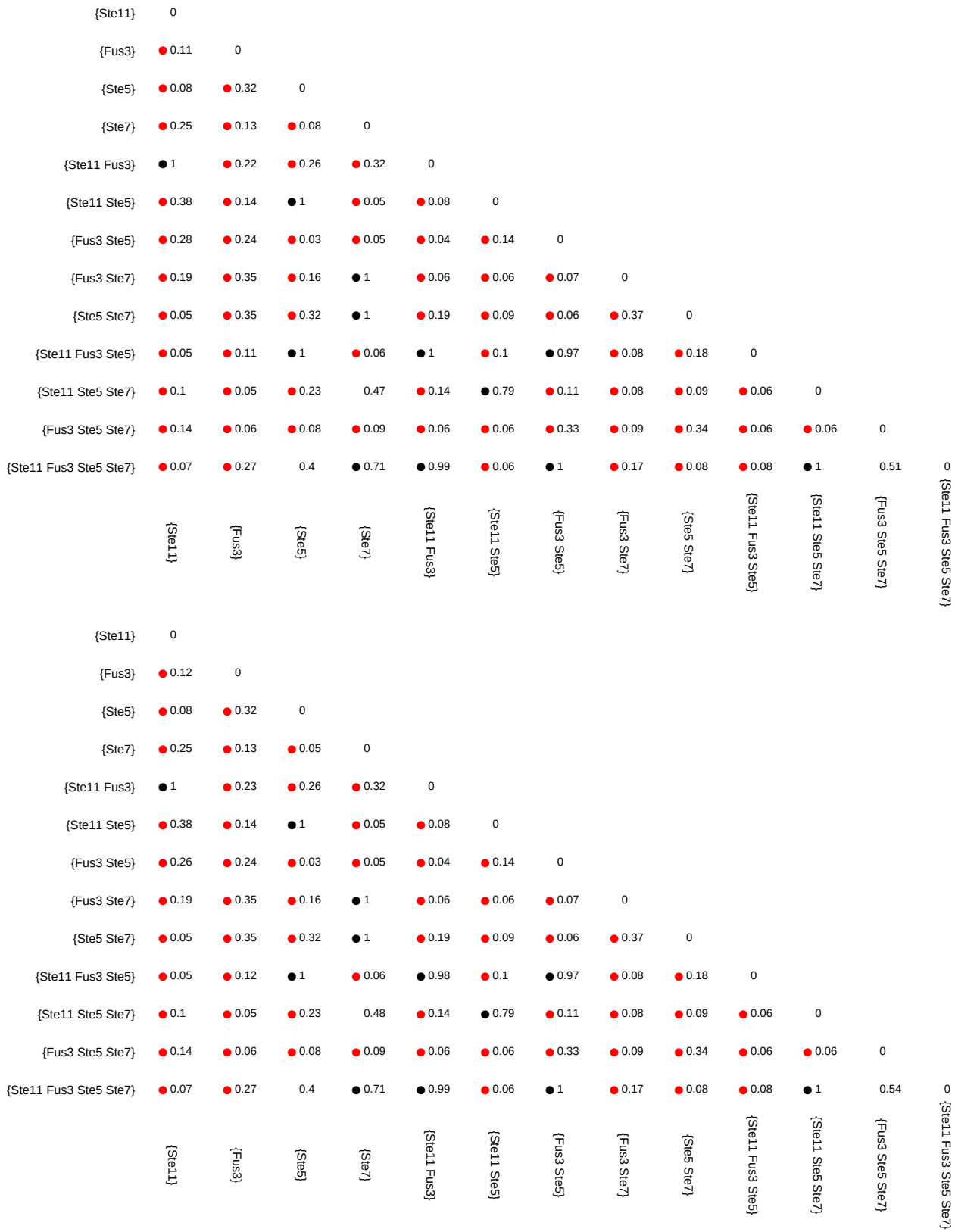


Abbildung 20: *Geschätzte pep-Matrix für die lang stimulierte Hefe, berechnet mit NPBN-DP (oben) und NPBN-PY(unten).*

Reagierende Spezies	Reaktionsrichtung	Reagierende Spezies
c1.pro_EGFR	$\longleftrightarrow$	compartment.EGFR
compartment.EGF + compartment.EGFR	$\longleftrightarrow$	compartment.L_EGFR
compartment.L_EGFR	$\longleftrightarrow$	compartment.L_EGFR_dimer
c1.SOS + c1.Grb2	$\longleftrightarrow$	c1.SOS_Grb2
c1.Grb2 + c1.pSOS	$\longleftrightarrow$	c1.pSOS_Grb2
compartment.L_EGFR_dimer	$\longleftrightarrow$	compartment.L_dpEGFR
compartment.L_dpEGFR + c1.c_Cbl	$\longleftrightarrow$	c1.dpEGFR_c_Cbl
compartment.L_dpEGFR + c1.pShc	$\longleftrightarrow$	c1.pShc_dpEGFR
c1.pDok	$\longleftrightarrow$	c1.Dok
c1.c_Cbl + c1.pShc_dpEGFR	$\longleftrightarrow$	c1.pShc_dpEGFR_c_Cbl
c1.pSOS	$\longrightarrow$	c1.SOS
c1.pSOS_Grb2	$\longrightarrow$	c1.SOS_Grb2
compartment.L_dpEGFR + c1.Shc	$\longleftrightarrow$	c1.Shc_dpEGFR
c1.Shc_dpEGFR	$\longrightarrow$	c1.pShc_dpEGFR
c1.dpEGFR_c_Cbl	$\longrightarrow$	c1.dpEGFR_c_Cbl_ubiq
c1.dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl
c1.c_Cbl + c1.Shc_dpEGFR	$\longleftrightarrow$	c1.Shc_dpEGFR_c_Cbl
c1.Shc_dpEGFR_c_Cbl	$\longrightarrow$	c1.Shc_dpEGFR_c_Cbl_ubiq
c1.Shc_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.Shc
c1.pShc_dpEGFR_c_Cbl	$\longrightarrow$	c1.pShc_dpEGFR_c_Cbl_ubiq
c1.pShc_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.pShc
c1.Shc_dpEGFR_c_Cbl	$\longrightarrow$	c1.pShc_dpEGFR_c_Cbl
c1.pShc + c1.SOS_Grb2	$\longleftrightarrow$	c1.Grb2_SOS_pShc
compartment.L_dpEGFR + c1.Grb2_SOS_pShc	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR
c1.pShc_dpEGFR + c1.SOS_Grb2	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR
c1.c_Cbl + c1.Grb2_SOS_pShc_dpEGFR	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
c1.dpEGFR_c_Cbl + c1.Grb2_SOS_pShc	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
c1.Grb2_SOS_pShc_dpEGFR_c_Cbl	$\longrightarrow$	c1.Grb2_SOS_pShc_dpEGFR_c_Cbl_ubiq
c1.Grb2_SOS_pShc_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.Grb2_SOS_pShc
c1.Grb2_SOS_pShc	$\longrightarrow$	c1.Shc + c1.SOS_Grb2
c1.Dok	$\longrightarrow$	c1.pDok + compartment.L_dpEGFR
+ compartment.L_dpEGFR		+ c1.Shc_dpEGFR + c1.pShc_dpEGFR
+ c1.Shc_dpEGFR		+ c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
+ c1.pShc_dpEGFR		+ c1.Grb2_SOS_pShc_dpEGFR
+ c1.Grb2_SOS_pShc_dpEGFR_c_Cbl		+ c1.dpEGFR_c_Cbl
+ c1.Grb2_SOS_pShc_dpEGFR		+ c1.Shc_dpEGFR_c_Cbl
+ c1.dpEGFR_c_Cbl + c1.Shc_dpEGFR_c_Cbl		+ c1.pShc_dpEGFR_c_Cbl
+ c1.pShc_dpEGFR_c_Cbl		+ c1.FRS2_dpEGFR
+ c1.FRS2_dpEGFR + c1.pFRS2_dpEGFR		+ c1.pFRS2_dpEGFR
+ c1.Crk_C3G_pFRS2_dpEGFR		+ c1.Crk_C3G_pFRS2_dpEGFR
+ c1.FRS2_dpEGFR_c_Cbl		+ c1.FRS2_dpEGFR_c_Cbl
+ c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl		+ c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl
c1.pShc	$\longrightarrow$	c1.Shc

Diese Tabelle wird auf der folgenden Seite fortgesetzt.

Reagierende Spezies	Reaktionsrichtung	Reagierende Spezies
c1.pFRS2	$\longrightarrow$	c1.FRS2
c1.Crk + c1.C3G	$\longleftrightarrow$	c1.Crk_C3G
compartment.L_dpEGFR + c1.FRS2	$\longleftrightarrow$	c1.FRS2_dpEGFR
compartment.L_dpEGFR + c1.pFRS2	$\longleftrightarrow$	c1.pFRS2_dpEGFR
c1.FRS2_dpEGFR	$\longrightarrow$	c1.pFRS2_dpEGFR
c1.pFRS2_dpEGFR + c1.Crk_C3G	$\longleftrightarrow$	c1.Crk_C3G_pFRS2_dpEGFR
c1.FRS2_dpEGFR + c1.c_Cbl	$\longleftrightarrow$	c1.FRS2_dpEGFR_c_Cbl
c1.c_Cbl + c1.pFRS2_dpEGFR	$\longleftrightarrow$	c1.pFRS2_dpEGFR_c_Cbl
c1.pFRS2_dpEGFR_c_Cbl	$\longrightarrow$	c1.pFRS2_dpEGFR_c_Cbl_ubiq
c1.FRS2_dpEGFR_c_Cbl	$\longrightarrow$	c1.FRS2_dpEGFR_c_Cbl_ubiq
c1.FRS2_dpEGFR_c_Cbl	$\longrightarrow$	c1.pFRS2_dpEGFR_c_Cbl
c1.pFRS2_dpEGFR_c_Cbl + c1.Crk_C3G	$\longleftrightarrow$	c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl
c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl	$\longrightarrow$	c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl_ubiq
c1.FRS2_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.FRS2
c1.pFRS2_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.pFRS2
c1.pDok + c1.RasGAP	$\longleftrightarrow$	c1.pDok_RasGAP
c1.SOS_Grb2 + c1.dppERK	$\longrightarrow$	c1.pSOS_Grb2 + c1.dppERK
c1.SOS + c1.dppERK	$\longrightarrow$	c1.pSOS + c1.dppERK
c1.c_Raf + c1.Ras_GTP	$\longleftrightarrow$	c1.c_Raf_Ras_GTP
c1.Rap1_GTP + c1.B_Raf	$\longleftrightarrow$	c1.B_Raf_Rap1_GTP
c1.Ras_GTP + c1.B_Raf	$\longleftrightarrow$	c1.B_Raf_Ras_GTP
c1.ppMEK + c1.PP2A	$\longrightarrow$	c1.pMEK + c1.PP2A
c1.pMEK + c1.PP2A	$\longrightarrow$	c1.MEK + c1.PP2A
c1.pMEK_ERK + c1.PP2A	$\longrightarrow$	c1.MEK_ERK + c1.PP2
2 c1.ppERK	$\longleftrightarrow$	c1.dppERK
c1.Ras_GTP	$\longrightarrow$	c1.Ras_GDP
c1.Rap1_GTP	$\longrightarrow$	c1.Rap1_GDP
c1.Rap1_GDP	$\longrightarrow$	c1.Rap1_GTP
+ c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl		+ c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl
+ c1.Crk_C3G_pFRS2_dpEGFR		+ c1.Crk_C3G_pFRS2_dpEGFR
+ c1.Crk_C3G_pFRS2_pTrkA_endo		
c1.Crk_C3G_pFRS2_pTrkA_endo c1.Ras_GDP	$\longrightarrow$	c1.Ras_GTP
+ c1.Grb2_SOS_pShc_dpEGFR_c_Cbl		+ c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
+ c1.Grb2_SOS_pShc_dpEGFR +		+ c1.Grb2_SOS_pShc_dpEGFR
c1.Grb2_SOS_pShc_pTrkA		+ c1.Grb2_SOS_pShc_pTrkA
compartment.NGF + compartment.NGFR	$\longleftrightarrow$	compartment.L_NGFR
compartment.L_NGFR	$\longrightarrow$	compartment.pTrkA
compartment.pTrkA	$\longrightarrow$	c1.pTrkA_endo
c1.pTrkA_endo	$\longrightarrow$	c1.degradation
compartment.pTrkA	$\longrightarrow$	c1.degradation
c1.Shc + compartment.pTrkA	$\longleftrightarrow$	c1.Shc_pTrkA
c1.pShc + compartment.pTrkA	$\longleftrightarrow$	c1.pShc_pTrkA
c1.FRS2 + compartment.pTrkA	$\longleftrightarrow$	c1.FRS2_pTrkA
c1.pFRS2 + compartment.pTrkA	$\longleftrightarrow$	c1.pFRS2_pTrkA

Diese Tabelle wird auf der folgenden Seite fortgesetzt.

Reagierende Spezies	Reaktionsrichtung	Reagierende Spezies
c1.pTrkA_endo + c1.Shc	$\longleftrightarrow$	c1.Shc_pTrkA_endo
c1.pTrkA_endo + c1.pShc	$\longleftrightarrow$	c1.pShc_pTrkA_endo
c1.FRS2_pTrkA	$\longrightarrow$	c1.pFRS2_pTrkA
c1.pTrkA_endo + c1.FRS2	$\longleftrightarrow$	c1.FRS2_pTrkA_endo
c1.pTrkA_endo + c1.pFRS2	$\longleftrightarrow$	c1.pFRS2_pTrkA_endo
c1.FRS2_pTrkA_endo	$\longrightarrow$	c1.pFRS2_pTrkA_endo
c1.FRS2_pTrkA	$\longrightarrow$	c1.degradation + c1.FRS2
c1.pFRS2_pTrkA	$\longrightarrow$	c1.degradation + c1.pFRS2
c1.Shc_pTrkA	$\longrightarrow$	c1.degradation + c1.Shc
c1.pShc_pTrkA	$\longrightarrow$	c1.degradation + c1.pShc
c1.FRS2_pTrkA_endo	$\longrightarrow$	c1.degradation + c1.FRS2
c1.Shc_pTrkA_endo	$\longrightarrow$	c1.degradation + c1.Shc
c1.pShc_pTrkA_endo	$\longrightarrow$	c1.degradation + c1.pShc
c1.SOS_Grb2 + c1.pShc_pTrkA_endo	$\longleftrightarrow$	c1.Grb2_SOS_pShc_pTrkA_endo
c1.c_Raf_Ras_GTP + c1.MEK	$\longrightarrow$	c1.c_Raf_Ras_GTP_MEK
c1.c_Raf_Ras_GTP + c1.MEK_ERK	$\longleftrightarrow$	c1.c_Raf_Ras_GTP_MEK_ERK
c1.c_Raf_Ras_GTP + c1.pDok_RasGAP	$\longrightarrow$	c1.c_Raf + c1.Ras_GDP + c1.pDok_RasGAP
c1.c_Raf_Ras_GTP + c1.pMEK	$\longleftrightarrow$	c1.c_Raf_Ras_GTP_pMEK
c1.c_Raf_Ras_GTP + c1.pMEK_ERK	$\longleftrightarrow$	c1.c_Raf_Ras_GTP_pMEK_ERK
c1.c_Raf_Ras_GTP_MEK	$\longrightarrow$	c1.c_Raf_Ras_GTP + c1.pMEK
c1.c_Raf_Ras_GTP_MEK_ERK	$\longrightarrow$	c1.c_Raf_Ras_GTP + c1.pMEK_ERK
c1.c_Raf_Ras_GTP_pMEK	$\longrightarrow$	c1.c_Raf_Ras_GTP + c1.ppMEK
c1.c_Raf_Ras_GTP_pMEK_ERK	$\longrightarrow$	c1.c_Raf_Ras_GTP + c1.ppMEK_ERK
c1.dpEGFR_c_Cbl + c1.FRS2	$\longleftrightarrow$	c1.FRS2_dpEGFR_c_Cbl
c1.dpEGFR_c_Cbl + c1.Grb2_SOS_pShc	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
c1.dpEGFR_c_Cbl + c1.pFRS2	$\longleftrightarrow$	c1.pFRS2_dpEGFR_c_Cbl
c1.dpEGFR_c_Cbl + c1.pShc	$\longleftrightarrow$	c1.pShc_dpEGFR_c_Cbl
c1.dpEGFR_c_Cbl	$\longrightarrow$	c1.dpEGFR_c_Cbl_ubiq
c1.dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl
c1.dppERK_MKP3	$\longrightarrow$	c1.ppERK + c1.ERK + c1.MKP3
c1.pDok + c1.RasGAP	$\longleftrightarrow$	c1.pDok_RasGAP
c1.pDok	$\longleftrightarrow$	c1.Dok
c1.pFRS2 + compartment.pTrkA	$\longleftrightarrow$	c1.pFRS2_pTrkA
c1.pFRS2	$\longrightarrow$	c1.FRS2
c1.pFRS2_dpEGFR + c1.Crk_C3G	$\longleftrightarrow$	c1.Crk_C3G_pFRS2_dpEGFR
c1.pFRS2_dpEGFR_c_Cbl + c1.Crk_C3G	$\longleftrightarrow$	c1.Crk_C3G_pFRS2_dpEGFR_c_Cbl
c1.pFRS2_dpEGFR_c_Cbl	$\longrightarrow$	c1.pFRS2_dpEGFR_c_Cbl_ubiq
c1.pFRS2_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.pFRS2
c1.pFRS_pTrkA	$\longrightarrow$	c1.degradation + c1.pFRS2
c1.pFRS2_pTrkA	$\longrightarrow$	c1.pFRS2_pTrkA_endo
c1.pFRS2_pTrkA_endo	$\longrightarrow$	c1.degradation + c1.pFRS2
c1.pMEK + c1.PP2A	$\longrightarrow$	c1.MEK + c1.PP2A
c1.pMEK_ERK + c1.PP2A	$\longrightarrow$	c1.MEK_ERK + c1.PP2A
c1.pSOS	$\longrightarrow$	c1.SOS

Diese Tabelle wird auf der folgenden Seite fortgesetzt.

Reagierende Spezies	Reaktionsrichtung	Reagierende Spezies
c1.pSOS_Grb2	$\longrightarrow$	c1.SOS_Grb2
c1.pShc + c1.SOS_Grb2	$\longleftrightarrow$	c1.Grb2_SOS_pShc
c1.pShc + compartment.pTrkA	$\longleftrightarrow$	c1.pShc_pTrkA
c1.pShc	$\longrightarrow$	c1.Shc
c1.pShc_dpEGFR + c1.SOS_Grb2	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR
c1.pShc_dpEGFR_c_Cbl + c1.SOS_Grb2	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR_c_Cbl
c1.pShc_dpEGFR_c_Cbl	$\longrightarrow$	c1.pShc_dpEGFR_c_Cbl_ubiq
c1.pShc_dpEGFR_c_Cbl_ubiq	$\longrightarrow$	c1.proteasome + c1.c_Cbl + c1.pShc
c1.pShc_pTrkA	$\longrightarrow$	c1.degradation + c1.pShc
c1.pShc_pTrkA	$\longrightarrow$	c1.pShc_pTrkA_endo
c1.pShc_pTrkA_endo	$\longrightarrow$	c1.degradation + c1.pShc
c1.pTrkA_endo + c1.FRS2	$\longleftrightarrow$	c1.FRS2_pTrkA_endo
c1.pTrkA_endo + c1.Shc	$\longleftrightarrow$	c1.Shc_pTrkA_endo
c1.pTrkA_endo + c1.pFRS2	$\longleftrightarrow$	c1.pFRS2_pTrkA_endo
c1.pTrkA_endo + c1.pShc	$\longleftrightarrow$	c1.pShc_pTrkA_endo
c1.pTrkA_endo	$\longrightarrow$	c1.degradation
c1.ppERK_MKP3	$\longrightarrow$	c1.ERK + c1.MKP3
c1.ppMEK + c1.PP2A	$\longrightarrow$	c1.pMEK + c1.PP2A
c1.ppMEK_ERK + c1.PP2A	$\longrightarrow$	c1.pMEK_ERK + c1.PP2A
c1.ppMEK_ERK	$\longrightarrow$	c1.ppERK + c1.ppMEK
c1.pro_EGFR	$\longleftrightarrow$	compartment.EGFR
c1.pro_TrkA	$\longleftrightarrow$	compartment.NGFR
compartment.EGF + compartment.EGFR	$\longleftrightarrow$	compartment.L_EGFR
compartment.L_EGFR_dimer	$\longleftrightarrow$	compartment.L_dpEGFR
compartment.L_NGFR	$\longrightarrow$	compartment.pTrkA
compartment.L_dpEGFR + c1.FRS2	$\longleftrightarrow$	c1.FRS2_dpEGFR
compartment.L_dpEGFR + c1.Grb2_SOS_pShc	$\longleftrightarrow$	c1.Grb2_SOS_pShc_dpEGFR
compartment.L_dpEGFR + c1.Shc	$\longleftrightarrow$	c1.Shc_dpEGFR
compartment.L_dpEGFR + c1.c_Cbl	$\longleftrightarrow$	c1.dpEGFR_c_Cbl
compartment.L_dpEGFR + c1.pFRS2	$\longleftrightarrow$	c1.pFRS2_dpEGFR
compartment.L_dpEGFR + c1.pShc	$\longleftrightarrow$	c1.pShc_dpEGFR
compartment.NGF + compartment.NGFR	$\longleftrightarrow$	compartment.L_NGFR
compartment.pTrkA	$\longrightarrow$	c1.degradation
compartment.pTrkA	$\longrightarrow$	c1.pTrkA_endo

Tabelle 6: *Übersicht der in der Implementierung des Erk-Signalübertragungsnetzwerk-Modells nach Sasagawa et al. (2005) modellierten chemischen Reaktionen. Dabei kodiert das Symbol „$\longrightarrow$“ eine Reaktion, in der die Spezies links des Symbols zu den Spezies auf der rechten Seite des Symbols umgesetzt werden. Steht in einer Zeile hingegen das Symbol „$\longleftrightarrow$“ bedeutet dies, dass die beschriebene chemische Reaktion in beide Richtungen ablaufen kann. Das Symbol „+“ steht in chemischen Reaktionsgleichungen für „und“.*

{Ste11}	{Fus3}	{Ste5}	{Ste7}	{Ste11 Fus3}	{Ste11 Ste5}	{Ste11 Ste7}	{Fus3 Ste5}	{Fus3 Ste7}	{Ste5 Ste7}	{Ste11 Fus3 Ste5}	{Ste11 Fus3 Ste7}	{Ste11 Ste5 Ste7}	{Fus3 Ste5 Ste7}	{Ste11 Fus3 Ste5 Ste7}	$= \boldsymbol{\beta}$ $= \mathbf{y}$
0	0	0	1	0	0	1	0	1	1	0	1	1	1	1	AC Ste7
0	1	0	0	1	0	0	1	1	0	1	1	0	1	1	AC Fus3
0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	CC Ste7 Fus3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	0	0	0	1	1	1	0	0	0	1	1	1	0	1	AC Ste11
0	1	0	0	1	0	0	1	1	0	1	1	0	1	1	AC Fus3
0	0	0	0	1	0	0	0	0	0	1	1	0	0	1	CC Ste11 Fus3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	0	0	0	1	1	1	0	0	0	1	1	1	0	1	AC Ste11
0	0	0	1	0	0	1	0	1	1	0	1	1	1	1	AC Ste7
0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	CC Ste11 Ste7
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	0	0	0	1	1	1	0	0	0	1	1	1	0	1	AC Ste11
0	0	1	0	0	1	0	1	0	1	1	0	1	1	1	AC Ste5
0	0	0	0	0	1	0	0	0	0	1	0	1	0	1	CC Ste11 Ste5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	1	0	0	1	0	0	1	1	0	1	1	0	1	1	AC Fus3
0	0	1	0	0	1	0	1	0	1	1	0	1	1	1	AC Ste5
0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	CC Fus3 Ste5
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	1	0	0	1	0	1	0	1	1	0	1	1	1	AC Ste5
0	0	0	1	0	0	1	0	1	1	0	1	1	1	1	AC Ste7
0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	CC Ste5 Ste7
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Tabelle 7: *Modellbestandteile des Schätzers der Konzentration der Proteinkomplexe aus Kapitel 5.3. Die Beschreibung wird auf der folgenden Seite fortgesetzt.*

In den Spalten 1-15, ist die Designmatrix $\mathbf{X}$ *abgebildet. Die aus drei Zeilen bestehenden Blöcke wiederholen sich für jede FCS-Messung. Die Spalten geben die gesuchten Konzentrationen der Proteinkomplexe an. Die letzte Spalte enthält den Vektor der gemessenen Konzentrationen der markierten Proteine. Jede neue Messung erweitert den Vektor um drei Einträge. Das Nichtvorkommen der Komplexe* $\{$*Ste11 Ste7* $\}$ *und* $\{$*Ste11 Fus3 Ste7* $\}$ *wird in der Schätzung durch Nebenbedingungen auf dem Vektor* $\boldsymbol{\beta}$ *berücksichtigt.*

G. Konvergenzdiagnostik

In dieser Arbeit wird das Hauptaugenmerk auf die Analyse der gefundenen Netzwerke und die Entmischung gelegt. Der entwickelte Sampler operiert im Raum der DAGs und im Raum der Allokationsvektoren. Beide Größen sind nicht stetig und aufgrund ihrer Struktur aufwändig in der Handhabung, so dass eine direkte Überprüfung der Konvergenz durch Standarddiagnoseverfahren nicht möglich ist.

Konvergenzdiagnostik der Graphenstruktur

Für die Überprüfung der Konvergenz der Netzwerke wird in Grzegorczyk (2011) vorgeschlagen, die Konvergenz der Graphen anhand der Konvergenz der Kanten zu überprüfen. Nach Entfernung des Burn-Ins wird die MCMC-Kette der DAGs in mehrere gleichlange Abschnitte aufgeteilt und in jedem Abschnitt werden die a posteriori Kantenwahrscheinlichkeiten bestimmt, vergleiche Abschnitt 3.3.1. Dann kann die Konvergenz für jede Kanten-Kette separat überprüft werden. Hierzu werden die von Giudici und Castelo (2003) vorgeschlagenen trace plots und die Geweke-Diagnostik (Geweke, 1992) betrachtet. Für eine bessere Unterscheidbarkeit werden die Ketten in unterschiedlichen Farben abgebildet. Denkbar wäre auch die Verwendung alternativer Diagnoseverfahren, welche mehrere MCMC Ketten vergleichen wie z.B. der potential scale reduction factor von Brooks und Gelman (1998). Diese Methoden sind aber verglichen mit dem gewählten Verfahren um ein vielfaches rechenintensiver. Von ihrer Anwendung wird daher abgesehen.

Konvergenzdiagnostik des Allokationsvektors

Üblicherweise wird die Konvergenzdiagnostik des Allokationsvektors anhand von abgeleiteten Größen vorgenommen (Grzegorczyk et al., 2008). Die logarithmierten BGe Scores bieten sich hierfür an, da davon ausgegangen werden kann, dass, wenn die Kette der Graphen die stationäre Verteilung erreicht hat, das Verhalten des BGe Scores nur noch von dem Allokationsvektor abhängt. Die Betrachtung beider Größen (der Kantenwahr-

scheinlichkeiten und des BGe Scores) ermöglicht eine umfassende Konvergenzdiagnose des NPBN-Schätzers.

Diagnoseergebnisse ausgewählter Netzwerke

Aus Platzgründen werden die Ergebnisse der durchgeführten Diagnosen nur für einige Beispiele wiedergegeben. Von den über 140 ausgewerteten Ketten der Simulationsstudie werden bevorzugt die Netzwerke mit hoher Noisestärke betrachtet, da davon ausgegangen werden kann, dass die Netzwerke, welche weniger stark mit Rauschen belegt sind, deutlich schneller konvergieren. Die Ketten aus der Analyse des Hefenetzwerkes in Kapitel 5 werden ebenfalls untersucht.

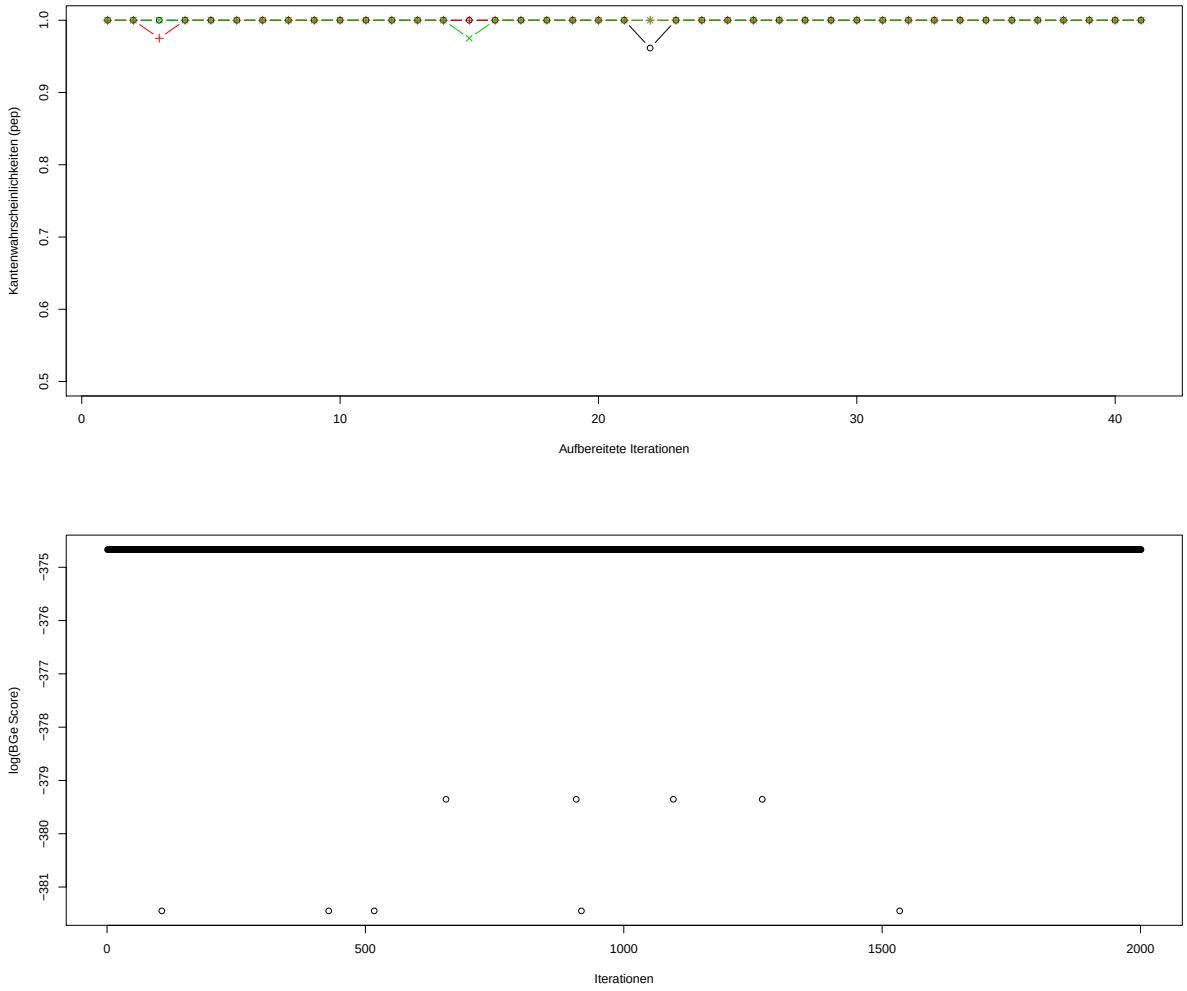


Abbildung 21: *Traceplots des simulierten Netzwerks mit vier Komponenten und Noise 0.1 zum Zeitpunkt 1 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.32, 0.33, 0.33 (oben), der BGe Score mit p-Wert 0.62 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

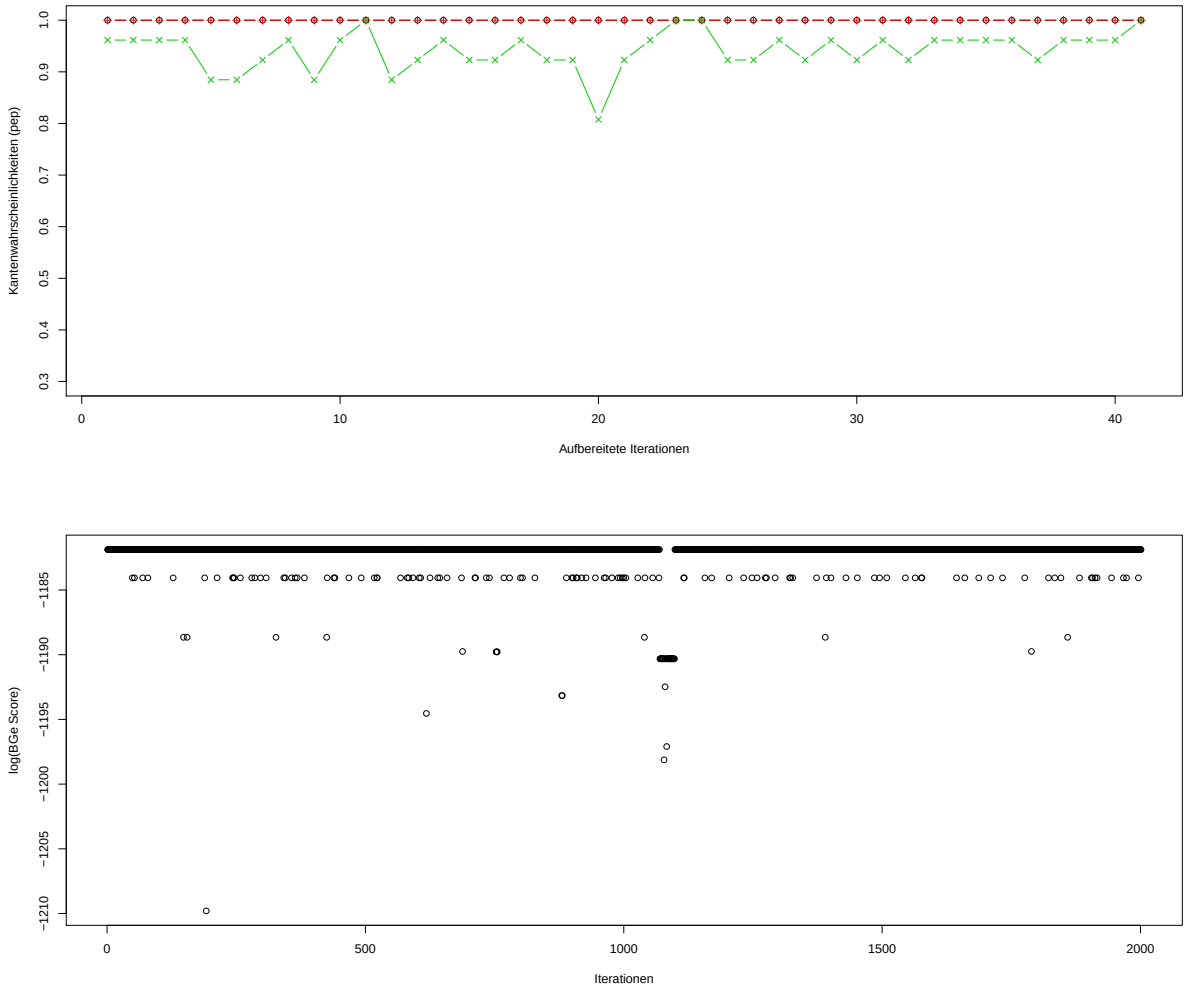


Abbildung 22: *Traceplots des simulierten Netzwerks mit vier Komponenten und Noise 0.2 zum Zeitpunkt 8 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.32, 0.32, 0.32 (oben), der BGe Score mit p-Wert 0.73 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

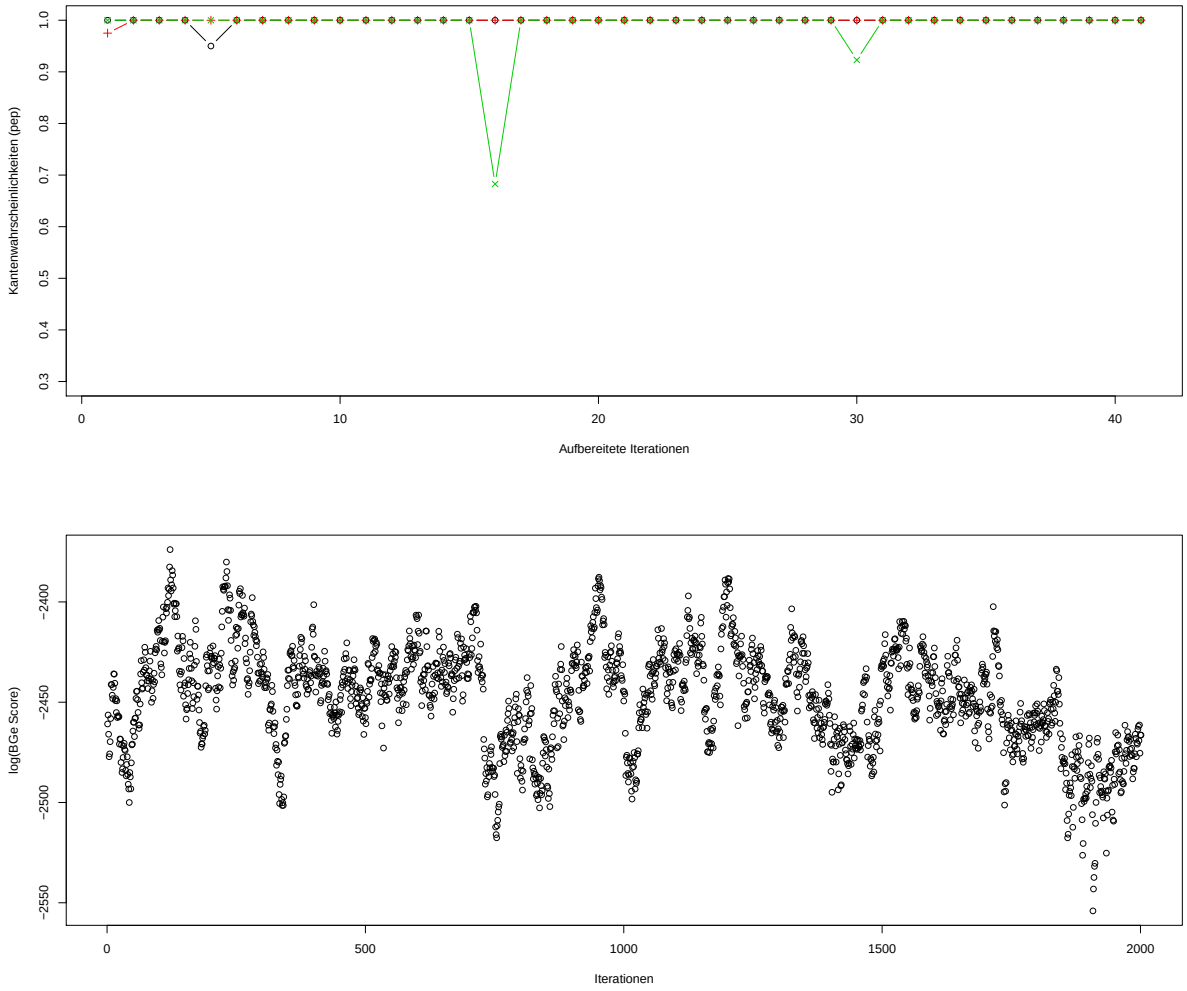


Abbildung 23: *Traceplots des simulierten Netzwerks mit vier Komponenten und Noise 0.4 zum Zeitpunkt 6 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.32, 0.32, 0.31 (oben), der BGe Score mit p-Wert 0.28 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

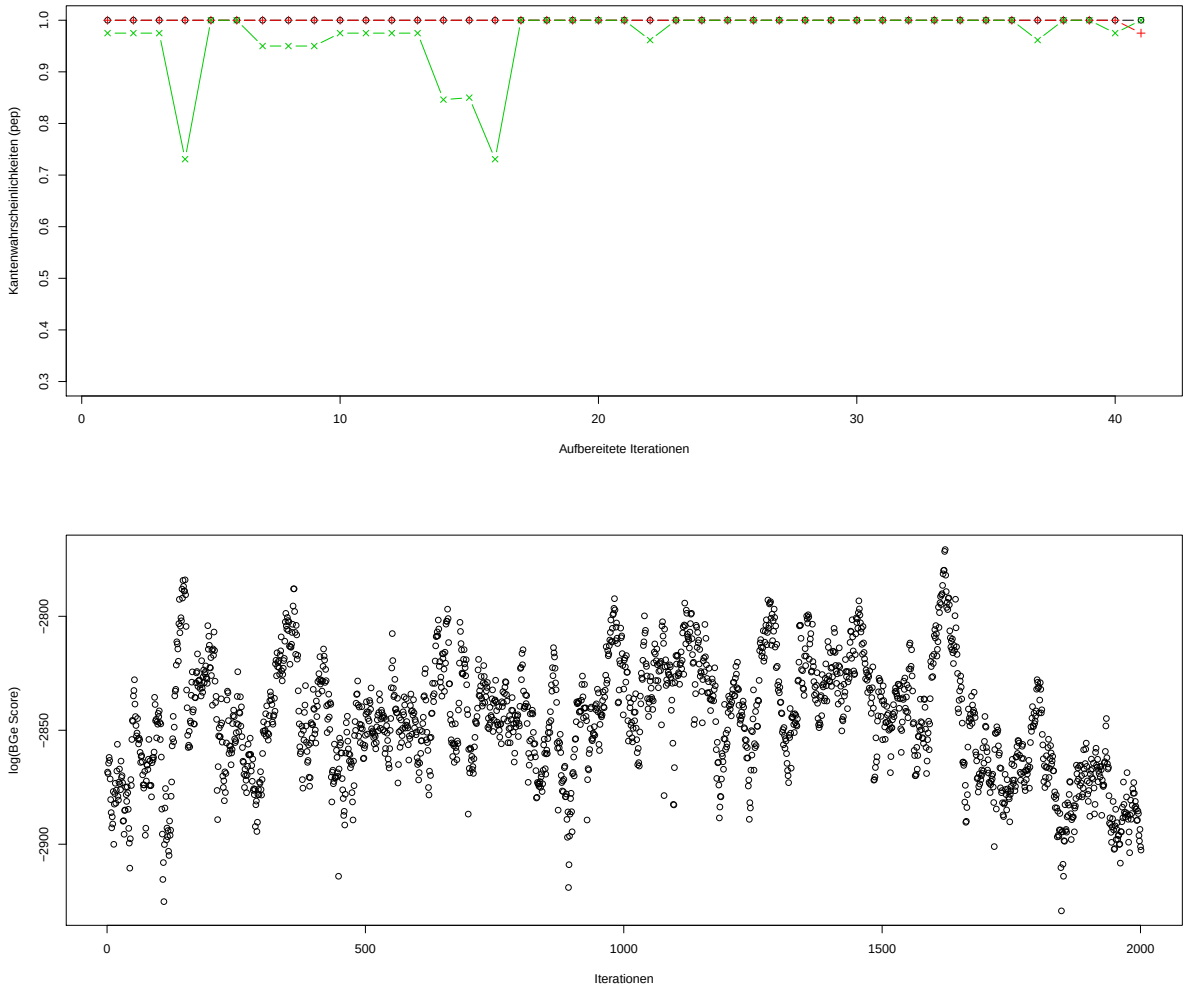


Abbildung 24: *Traceplots des simulierten Netzwerks mit vier Komponenten und Noise 0.7 zum Zeitpunkt 3 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.32, 0.32, 0.20 (oben), der BGe Score mit p-Wert 0.54 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

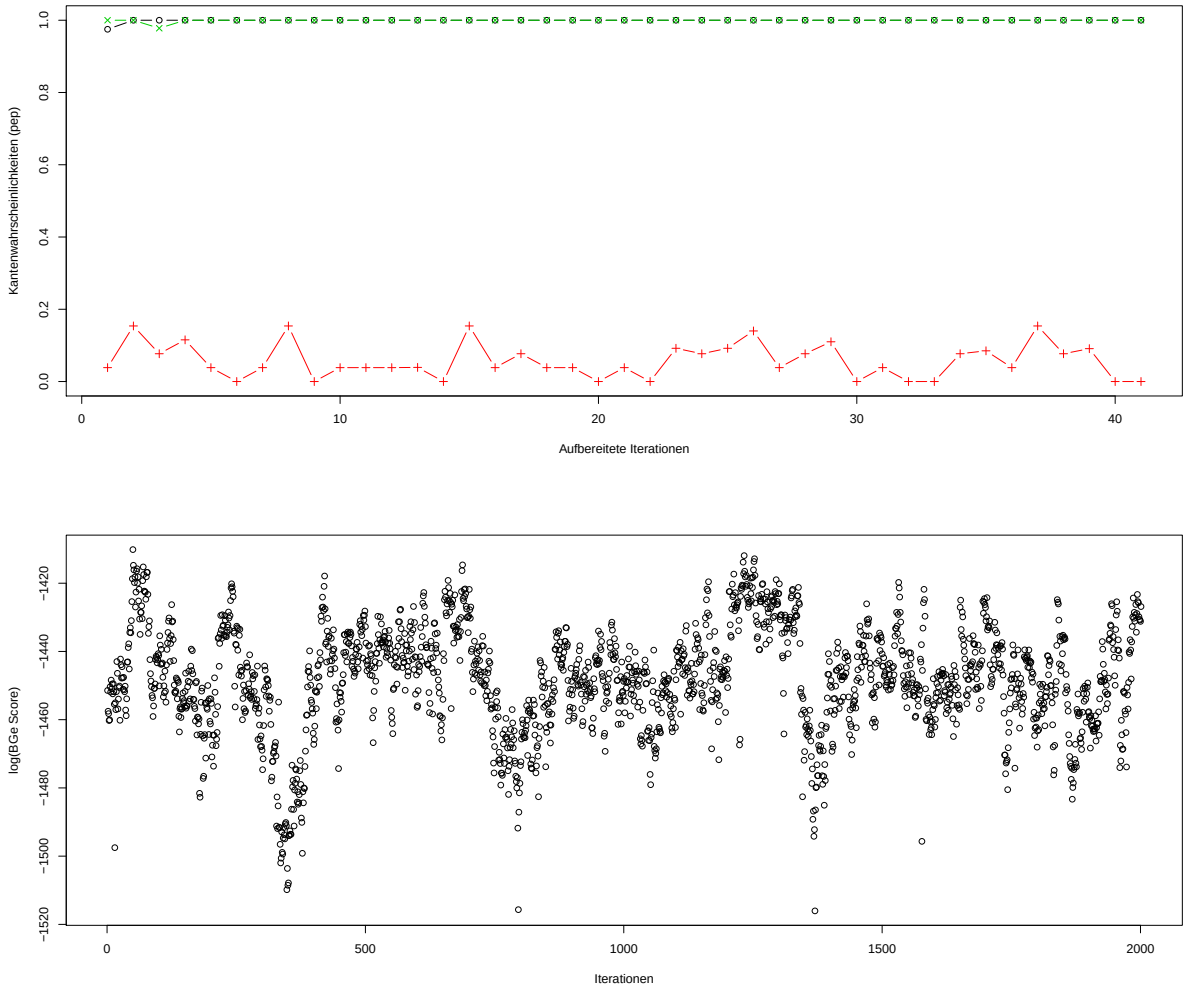


Abbildung 25: *Traceplots des simulierten Netzwerks mit zwei Komponenten und Noise 0.6 zum Zeitpunkt 2 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.32 0.27 0.32 (oben), der BGe Score mit p-Wert 0.82 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

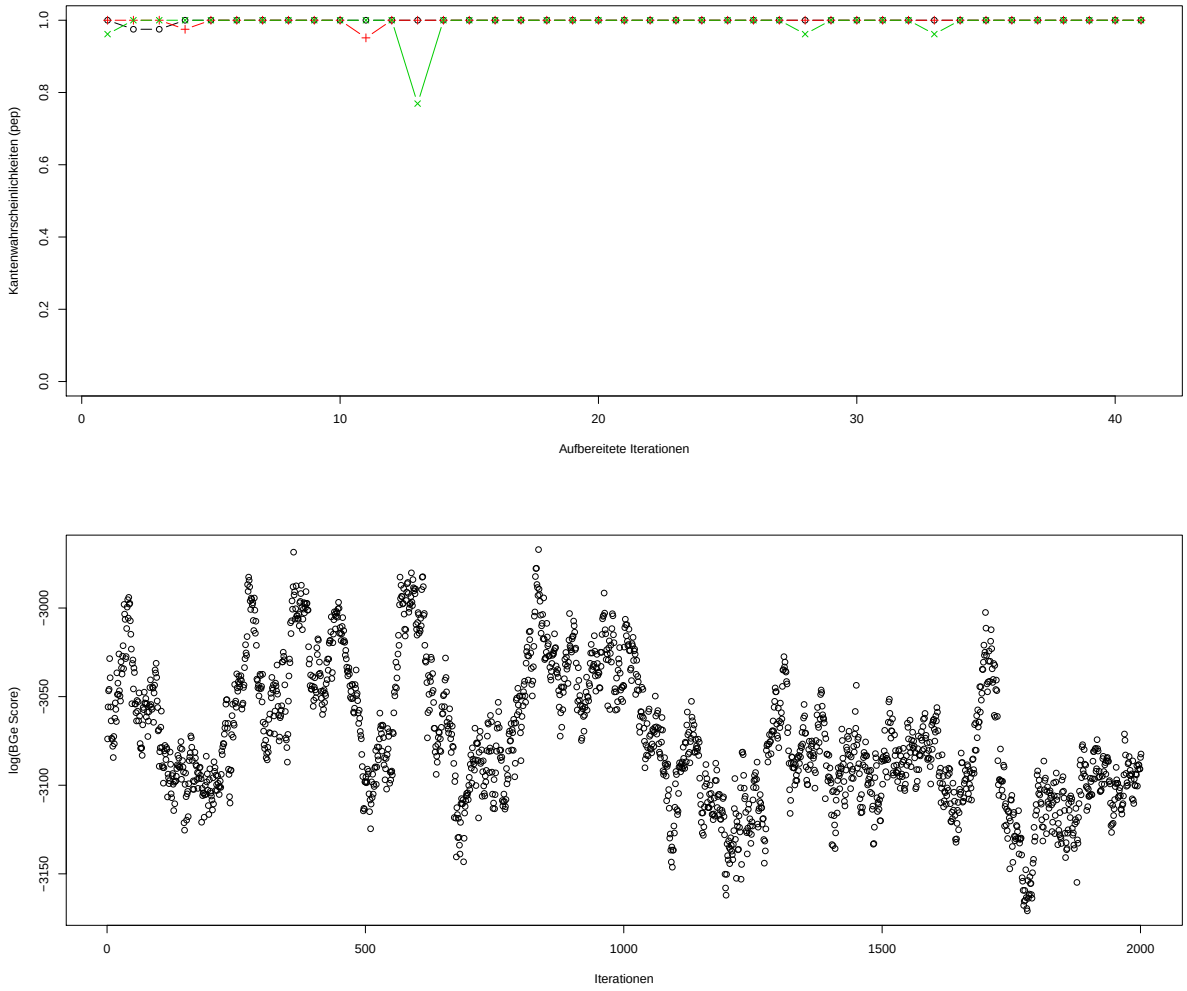


Abbildung 26: *Traceplots des simulierten Netzwerks mit vier Komponenten und Noise 0.7 zum Zeitpunkt 6 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Pitman-Yor a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.10 0.32 0.62 (oben), der BGe Score mit p-Wert 0.14 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

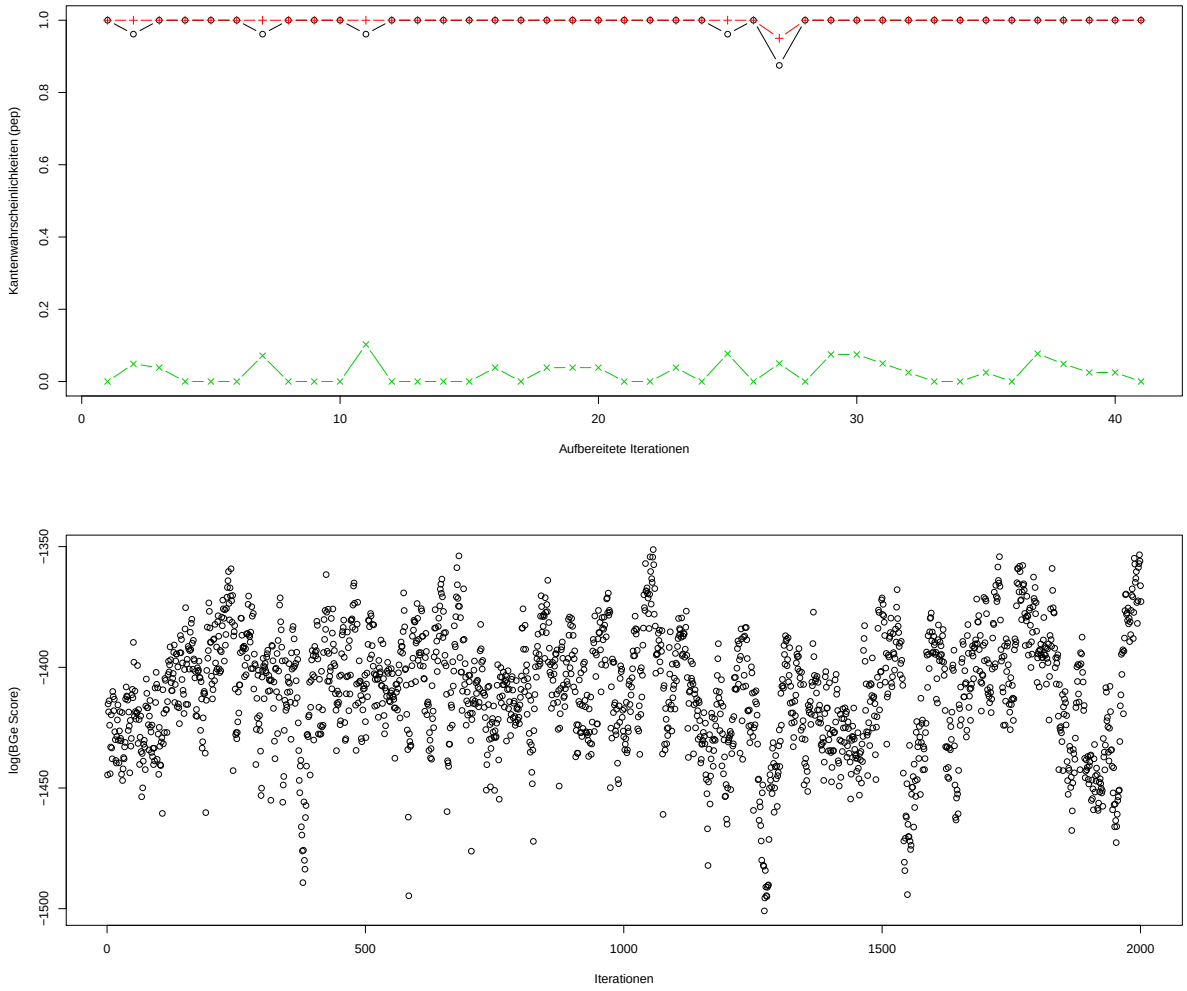


Abbildung 27: *Traceplots des simulierten Netzwerks mit zwei Komponenten und Noise 0.7 zum Zeitpunkt 10 Min. nach der Stimulation, analysiert mit dem NPBN-Verfahren unter Verwendung der Pitman-Yor a priori. pep-Werte mit den zugehörigen p-Werten der Geweke-Diagnostik: 0.79 0.32 0.40 (oben), der BGe Score mit p-Wert 0.70 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

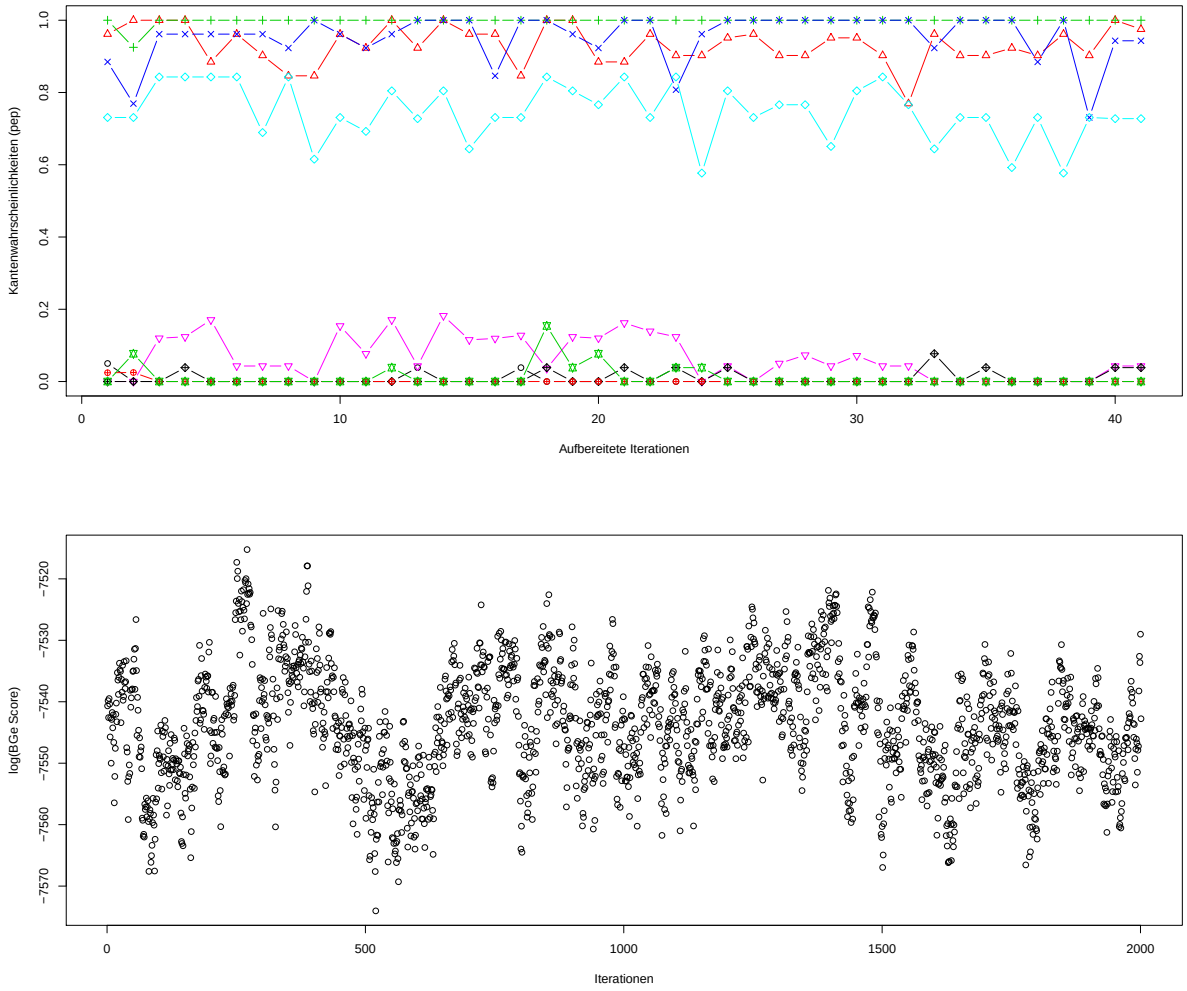


Abbildung 28: *Traceplots des Hefenetzwerks mit drei Komponenten, basierend auf den aufbereiteten FCS Messungen, analysiert mit dem NPBN-Verfahren unter Verwendung der Dirichlet a priori. pep-Werte mit den zugehörigen aufsteigend geordneten p-Werten der Geweke-Diagnostik: 0.05, 0.05, 0.06, 0.06, 0.07, 0.07, 0.07, 0.08, 0.09, 0.09, 0.10, 0.11, 0.11, 0.11, 0.12, 0.15, 0.17, 0.17, 0.17, 0.18, 0.18, 0.18, 0.20, 0.21, 0.23, 0.26, 0.26, 0.29, 0.29, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.37, 0.38, 0.39, 0.40, 0.40, 0.40, 0.41, 0.41, 0.41, 0.41, 0.42, 0.43, 0.44, 0.45, 0.45, 0.47, 0.47, 0.47, 0.47, 0.47, 0.48, 0.50, 0.50, 0.50, 0.51, 0.53, 0.62, 0.62, 0.62, 0.63, 0.65, 0.66, 0.69, 0.76, 0.78, 0.79, 0.91, 0.94, 0.97 (abgebildet wurden jeweils die fünf Ketten mit den höchsten und mit den niedrigsten Werten) (oben), der BGe Score mit p-Wert 0.70 (unten). In beiden Fällen kann die Nullhypothese „die Kette ist konvergiert“ nicht abgelehnt werden.*

H. Abkürzungs- und Stichwortverzeichnis

$\mathcal{A}$	Adjazenzmatrix
$\{ABC\}$	Molekülgebilde bestehend aus den Proteinen A,B und C
$AC(\wp)$	Autokorrelation von $\wp$
ASW	average silhouette width
BN	Bayessches Netzwerk 8
$\boldsymbol{\beta}$	Vektor der Unbekannten im Komplexeschätzer
$CC(\wp\wp^*)$	Kreuzkorrelation von $\wp$ und $\wp^*$
DAG	directed acyclic graph
E	Menge der Kanten im Graph $\mathcal{G}$ (Seite 7)
EPPF	exchangeable partition probability function
$\mathcal{G}$	Graph (Seite 7)
GBN	Gaußsches Bayessches Netzwerk
$\Gamma(a)$	Gammafunktion an der Stelle a
h	Index für die Anzahl der Gruppen (Seite 8)
i	Laufindex der Beobachtungen $i = 1, \ldots, n$
I_{j,j^*}	Mutual information für das Paar X_j und X_{j^*}
j	Laufindex der Knoten $j = 1, \ldots, d$
$\mathbf{K_j}$	Menge der Indizes der Elemente in pa(X_j) (Seite 17)
K_n	Anzahl der Gruppen in Abhängigkeit von der Anzahl der Beobachtungen
$\mathscr{K}_\psi$	Proteinkomplex
$\mathcal{K}_\psi$	Konzentration von $\mathscr{K}_w$
$\boldsymbol{l}$	Allokationsvektor (Seite 8)
$\mathcal{L}(\mathcal{G}\|\mathcal{X})$	Likelihood von $\mathcal{G}$ gegeben $\mathcal{X}$
MCMC	Markov Chain Monte Carlo
$\mathcal{M}_{\mathbb{X}}$	Raum der beschränkten endlichen Maße
μ	Maß aus $\mathcal{M}_{\mathbb{X}}$
$\mathbb{N}$	Menge der natürlichen Zahlen
$\mathrm{M}_{\mathrm{DAG}}$	Menge der Nachbargraphen von DAG

Nachbargraphen	DAGs die mittels einer einzigen Kantenoperation aus einen gegebenen DAG entstehen können
η	Noisestärke
NPBN	nichtparametrische Bayesschen Netzwerke
NPBN-DP	NPBN-Verfahren mit dem Dirichlet-Prozess als a priori
NPBN-PY	NPBN-Verfahren mit dem Pitman-Yor-Prozess als a priori
$\tilde{p}$	zufälliges Wahrscheinlichkeitsmaß (Seite 23)
$\mathbb{P}$	Wahrscheinlichkeitsmaß
P_0	Grundmaß
$\mathrm{pa}(X_j)$	Elternmenge von $\mathrm{pa}(X_j)$ (Seite 7)
$\wp$	Protein
pep-Matrix	Matrix der geschätzten Kantenwahrscheinlichkeiten
$\Pi_k^{(n)}$	EPPF einer austauschbaren Folge von Zufallsvariablen
$\mathbb{R}$	Menge der reellen Zahlen
ρ	partiellen Korrelation
$S(X_j)$	Entropie von X_j
Spezies	Objekte, die miteinander interagieren (systembiologischer Kontext)
Σ	Kovarianzmatrix
UNPBN	Unmixing via Nonparametric Bayesian Networks
V	Menge der Knoten im Graph $\mathcal{G}$ (Seite 7)
V_j	j-ter Knoten
$\boldsymbol{X}$	Designmatrix
X_j	Zufallsvariable assoziiert mit V_j
$\mathcal{X}$	Matrix der Beobachtungen (Seite 8)
$\vec{X}_{jj^*}$	gerichtete Kante
$\overset{\leftrightarrow}{X}_{jj^*}$	ungerichtete Kante
$\mathbf{y}$	Vektor der FCS Messungen (im Komplexeschätzer)
y_{zz^*}	FCS Messung bei markierten Proteinen $\wp_z$ und $\wp_{z^*}$
$\boldsymbol{Z} = (\mathcal{Z}_1, \ldots, \mathcal{Z}_n)$	Beobachtungen aus einem diskreten stochastischen Prozess