博士論文

科学技術文献における
引用ネットワークに対する確率生成モデル

安井 雄一郎

2023 年 3 月

総合研究大学院大学 複合科学研究科 統計科学専攻

謝辞

博士課程に在籍中，私を支えてくれた皆様に感謝申し上げます．主指導教官の金藤浩司教授には博士課程において多大な支援とご協力，忍耐強いご指導いただいたことを感謝いたします．また，前指導教官である中野純司教授に心から感謝します．中野教授には研究に関する議論に多くの時間を割いてくださいました．統計数理研究所の栗木哲教授，南和宏教授，東京工業大学の脇田建准教授に研究に関する有益なご助言をいただきました．感謝いたします．当時，総合研究大学院大学に在籍されていた Livia Lin-Hsuan Chang 氏には研究に関するご助言をいただきました．感謝いたします．Web of Science 書誌データベースの提供や，引用ネットワークのデータ公開の快諾をいただきましたクラリベイト・アナリティクス社に感謝いたします．書誌データを利用しやすくしてくださった統計数理研究所 URA チームの本多啓介氏と濵田ひろか氏に心から感謝いたします．また統計数理研究所の教職員の皆様，総合研究大学院大学の全ての学生に感謝します．最後にこの論文を執筆するために支えてくれた家族のさつき，梢，梓，渉に感謝いたします．

概要

本論文では学術論文や特許文献の引用ネットワーク構造に対する確率的生成モデルを提案する．

学術論文や特許文献の引用構造は，文献をノードに，文献間の引用構造を有向エッジに対応させた有向グラフで表現することができる．またノードには文献の発表時刻に対応する離散時刻が付与される．本研究では対象の引用ネットワークに対して簡素な生成プロセスを定義し，シミュレーションを用いた適合を検証することで，実データがもつ性質を明らかにすることを目的とする．このように実問題に対応するネットワーク構造を理解することで，対象の実問題の理解を深めることは，様々な先行研究で用いられる典型的なネットワーク科学の枠組みである．

本論文で提案するモデルは文献間の引用が，引用する文献の種類，引用された文献の重要度，両者の発表時刻の差にもとづく確率で発生することを想定する．ロジスティック関数に従う離散時間における文献数，逆ガウス確率分布の確率密度関数に従う発表時刻差ごとの引用のされやすさ，一般化パレート分布（または指数分布）に従う各文献における参考文献の件数などにもとづき，ネットワーク構造を生成する．その際にエッジの生成は，重要度に応じてノードを選択する優先的選択（Preferential attachment; PA）機構と，選択済みノードの隣接ノードを選択し三角形型の引用構造を構築する三角形形成（Triad formation; TF）機構を組み合わせる．なお文献の重要度と種類は，ノードの入次数と出次数で近似されるものとする．

まず学術論文の書誌データベース Web of Science における確率統計分野の文献から構築された引用ネットワークを対象とする．提案モデルは離散時刻ごとの文献数にもとづきノードを生成し，一定期間さかのぼった引用構造を対象にモデリングした時刻に調整された文献の年齢分布や出次数分布にもとづきエッジの生成を行う．シミュレーションにより，引用ネットワークで重要な特徴量となる，引用数の件数を表す入次数分布，被引用数の件数を表す出次数分布，三角形型の引用構造の件数を表す三角形数の分布に関して，モデルの適合を示した．さらに arXiv の書誌データベースにおける高エネルギー物理分野の文献から構築された二種類の引用ネットワークを対象に，モデル適合を検証した．これらは既存モデルでも検証に用いられる機会が多いネットワークである．また提案モデルは既存モデルと比較して，各ノードがもつ離散時刻を明示的に扱えること，実行時にデータを必要としない完全なシミュレーションを実現できることが利点である．

つづいて特許文献の引用構造から構築された引用ネットワークを対象とする．特許文献の引用ネットワークは各ノードがカテゴリやサブカテゴリにより分類され，階層的なクラスタ構造を形成している．カテゴリ内やサブカテゴリ内の引用は多く，カテゴリ間やサブカテゴリ間の引用は少ない．これは提案モデルの想定とは異なるため，あるサブカテゴリに着目して生成モデルの適用したものの，三

角形数の分布に対する適合に課題があることが明らかになった．そこで提案モデルの TF 機構に対する二種類の拡張を提案する．一点目としては TF 機構が選択するノードの候補の拡張である．提案モデルでは直前の PA 機構で選択されたノードの隣接ノードを候補に設定していた．これを基準となるノードから PA 機構により選択されたノード全ての隣接ノードへ候補を拡大した．二点目としては TF 機構を実行する割合のモデリングである．提案モデルの実行割合は定数パラメータで指定していたが，ノードごとに柔軟に設定できるように拡張を行った．まず各ノードにおける TF 機構の実行割合を予測するためのアルゴリズムを構成する．アルゴリズムは対象のネットワーク構造の各エッジが PA 機構もしくは TF 機構のどちらかで生成されたと仮定する．予測された TF 機構の実行割合の分布を観測することで，0 や 1 が過剰に存在すること，実行割合は実際によりも小さめに予測していること，ノードの出次数が大きいときに実行割合が大きくなることなどの特徴を有していることが明らかになった．そこで TF 機構の実行回数をゼロワン過剰ベータ分布でモデル化を行い，ある程度大きな出次数をもつノードでは実行割合を 1 に固定する．これらの拡張を適用した提案モデルは，特許文献の引用ネットワークや学術文献の引用ネットワークのどちらに対しても十分に適合することを確認できた．

本研究の貢献は主に引用ネットワークの各文献がもつ離散時間の考慮，シミュレーションからデータの直接的な使用の排除，ノードごとに異なる三角形形成機構の実行割合の制御である．改善モデルは 2 ノード間に関する特徴である次数，3 ノード間に関する特徴である三角形の性質はモデリングできたものの，ネットワークに関する広域な特徴量である Scree plot へのあてはまりは依然として十分でない．また引用ネットワーク全体を対象としたときに必要な階層構造のモデリングには至っていない．これらが残された課題として挙げられる．

目次

第 1 章　はじめに 2

第 2 章　引用ネットワークと提案された生成モデル 6

2.1 引用構造のネットワーク表現 6

2.2 ネットワーク特徴量 8

2.2.1 次数分布 8

2.2.2 ノードが参加する三角形数の分布 8

2.2.3 グラフ隣接行列に対する Scree plot 9

2.2.4 その他の指標 9

2.3 一般的なネットワーク生成モデル 11

2.3.1 Erdös–Rényi モデル 11

2.3.2 Watts–Strogatz モデル 11

2.3.3 *dk*-series モデル 11

2.3.4 Stochastic Kronecker Graph モデル 12

2.4 引用ネットワークに対する成長モデル 13

2.4.1 Barabási–Albert モデル 13

2.4.2 Holme–Kim モデル 14

2.4.3 コピーにもとづく生成モデル 14

2.4.4 Wu–Holme モデル 14

2.4.5 Chang–Phoa–Nakano モデル 15

2.5 引用ネットワークに対するモデル適応の確認 15

2.6 生成モデルとリンク予測 17

第 3 章　学術論文の引用ネットワークに対する確率生成モデル 18

3.1 引用ネットワーク `WoS-Stat` の構築 18

3.2 `WoS-Stat` の基本的な性質 21

3.3 時刻に依存したネットワーク特徴量 24

3.3.1 時間調整した引用の年齢分布 24

3.3.2 時間調整した出次数の分布 24

3.4 提案モデル 25
3.4.1 いくつか特徴量のモデル化 25
3.4.2 生成プロセスのモデル化 26
3.5 関数群の推定とシミュレーション 27
3.5.1 パラメータの推定 27
3.5.2 ネットワーク生成のためのシミュレーション 28
3.6 WoS-Stat に対するシミュレーションによるモデル適合の確認 31
3.6.1 各関数の推定結果の検討 31
3.6.2 パラメータ β の調整 31
3.6.3 ネットワーク特徴量を用いたモデル適合の確認 32
3.7 arXiv 引用ネットワークを用いた検証 35
3.7.1 引用ネットワークの構築 35
3.7.2 パラメータの推定 36
3.7.3 ネットワーク特徴量を用いたあてはまりの確認 36
3.8 シミュレーション結果の解釈 39

第 4 章 特許文献の引用ネットワークに対する確率生成モデル 40
4.1 特許文献の引用ネットワーク cit-Patents 40
4.1.1 カテゴリとサブカテゴリによる階層的なクラスタ構造 40
4.1.2 グラフ・クラスタリングを用いた階層構造の確認 42
4.1.3 カテゴリ構造とサブカテゴリ構造のネットワーク特徴量 45
4.2 引用ネットワークに対する生成モデルの適合の検証 49
4.3 各ノードにおける TF の実行割合のモデリング 51
4.3.1 各ノードにおける TF の実行回数の予測 51
4.4 特許文献の引用ネットワークに対する生成モデル 56
4.4.1 TF 機構が実行する割合のモデル化 56
4.4.2 生成プロセスのモデル化 56
4.4.3 3 章で提案したモデルとの比較 57
4.5 関数群の推定とシミュレーション 57
4.5.1 パラメータの推定 57
4.5.2 ネットワーク生成のためのシミュレーション 57
4.6 特許文献の引用ネットワークに対するシミュレーション 61
4.7 学術文献の引用ネットワークに対するシミュレーション 64
4.8 モデル拡張とシミュレーション結果の解釈 67

第 5 章 おわりに 68

参考文献 71

A 関連モデルのシミュレーションのためのアルゴリズム 76
A.1 初期化 76
A.2 Barabási–Albert モデル 76
A.3 Holme–Kim モデル 76
A.4 Wu–Holme モデル 78
B 本研究で用いた引用ネットワークの利用方法 79
B.1 `WoS-Stat` の利用方法 79
B.2 `arXiv-HepTh` と `arXiv-HepPh` における文献 ID から時刻情報の抽出 80
B.3 `cit-Patents` や `cit-Patents-sc41` の利用方法 82
B.4 提案モデルのシミュレーションのための参照実装 84

第 1 章

はじめに

学術論文は学術領域における主要な成果であり，近年，その件数は急激に増加している．そのため重要な文献を選出することは容易ではなく，文献の品質を評価することの重要性が高い．Impact factor [Garfield, 1955] や h-index [Hirsch, 2005] は，論文の質に基づいて学術雑誌と著者の質を評価するためによく知られた指標である．Institutional Research (IR) は学術文献に対する評価のあり方を研究する分野であり関心を集めている．IR における主要なトピックとして，論文の引用や共著など，論文の形式的な情報の分析が挙げられる．学術文献や科学技術文献における引用構造は，対象領域がこれまでにどのような発展をなされてきたのかをさかのぼることができ，理解を助ける有益な知識資源である．また引用構造が表現するこれまでの分野発展のされ方から，今後，どのような方向性へどのような発展を遂げるのか，その示唆が得られる可能性がある．

学術論文や特許文献の引用は引用ネットワーク (Citation network) と呼ばれる，簡素なネットワーク構造で表現することができる．文献をノード，文献間の引用を向きをもつエッジで対応させることで，引用や被引用をネットワーク構造上の特徴として扱うことができる．例えば被引用数の多い論文は，様々な論文に対して影響を与えた可能性が高く典型的に重要な論文とされる．ここである文献が引用する参考文献の件数はある対応するノードの出次数に，ある論文が引用された件数は対応するノードの入次数で対応できるが，これらはネットワーク構造を理解するための重要な特徴である．このように引用構造の分析を，ネットワーク上の問題に置き換えることが可能となる．なお重要性に関するより適した定義は文献 [Chang et al., 2019] などで議論されている．本研究では引用構造に興味をもち，対象の引用ネットワークと構造的に類似なネットワーク構造を生成するための確率的生成モデルを構築する．

利用可能な生成モデル

引用ネットワークに対する生成モデルは先行研究 [Price, 1976, Barabási and Albert, 1999, Holme and Kim, 2002, Wu and Holme, 2009, Leskovec et al., 2010, Chang et al., 2021] によって議論され

ており，ランダムグラフ生成器 (random graph generator) とも呼ばれる．生成モデルにおける成長は定義された機構に従いノードやエッジを追加して，ネットワーク構造を成長させる主要な方針である．優先的選択 (Preferential attachment; PA) [Barabási and Albert, 1999] はウェブページのリンク構造を対象として，より多くのウェブページからリンクされているウェブページはより多くのリンクを受け取る可能性が高いといった性質にもとづいた成長機構である．引用ネットワークにも応用されている [Price, 1976]．引用ネットワークでは引用がさらなる引用を呼ぶため，優先的選択で構築されるネットワーク構造に比べて，特に三角形型の引用が多い傾向があることを指摘された [Holme and Kim, 2002]．参考文献に含まれる文献間には引用構造が存在しやすいという性質を三角形形成機構としてモデリングされた．しかしながらこの引用の密集は，引用ネットワークにおける文献引用の大部分が引用した文献の参考文献を単にコピーしただけであると，引用構造の信頼性に問題視するといった指摘も存在する [Krapivsky and Redner, 2005]．さらに引用は文献間の引用の発表時刻の差によりに頻度が変化するといった，経年変化に関する機構も検討されている [Wu and Holme, 2009]．しかしながら発表時刻は，文献の順序にもとづいて表現されており，文献の発表時刻を明示的に扱っているわけではない．成長モデル以外では，小さなパラメータ行列に対するクロネッカー積を繰り返すことで生成されるクロネッカーグラフ [Leskovec et al., 2010] が挙げられる．クロネッカーグラフは少ないパラメータながら様々な種類のネットワーク構造を表現できる．

利用可能な引用ネットワーク

利用可能な引用ネットワークを挙げる．

- Physical Review Journals は Physical Review Letters, Physical Review, and Reviews of Modern Physics などから 1893 年以降の 45 万文献の書誌データ [*1] を含んでいる．
- Citation Network Dataset: DBLP+Citation, ACM Citation network [*2] では公開されているのは DBLP, ACM, MAG (Microsoft Academic Graph) とその他の情報ソースを用いて，構築された引用ネットワークである．2010-05-15 に公開された V1 は 629,814 文献と 632,752 引用だったのに対して，2021-05-14 に公開された V13 は 5,354,309 文献と 48,227,950 引用と大規模化している．
- The KONECT Project [*3] は様々な種類のネットワーク 1,326 件（2022 年 9 月 21 日時点）が収納されたレポジトリ [*4] を公開している．そのうち CiteSeerX から生成された引用ネットワーク (384,413 文献, 1,751,463 引用) や，機械学習文献の引用ネットワーク Cora (2708 文献，5429 引用) などの引用ネットワークを利用することができる．
- Semantic Scholar Academic Graph (S2AG) [Ammar et al., 2018] [*5] はアレン人工知能研究

[*1] https://journals.aps.org/datasets
[*2] https://www.aminer.org/citation
[*3] http://konect.cc/
[*4] http://konect.cc/networks
[*5] https://www.semanticscholar.org/

所で取り組まれている．自然言語処理を利用した要約や，HCI (human-computer interaction) による人間とコンピュータで協調した書誌データの整備などに 2022 年 9 月の時点で 2 億件以上の文献データを扱うことができる．

また引用ネットワークに関する文献関連サービスを挙げる．

- Web of Science [*6]は世界最大級の書誌データである．
- arXiv は文献をアップロード可能なウェブサイトであるが，API [*7] が公開されている．
- Google Scholar API [*8] は文献の検索や管理サービスで，API が利用可能で書誌データを抽出可能である．
- Microsoft Academic Graph [Sinha et al., 2015] [*9] は知識グラフとして，文献だけでない著者や研究トピックを統合的に構造化されている．文献間の類似度などの機能も利用できる．API [*10] も用意されている．
- uoa-eresearch [*11] はある研究者に関連する引用ネットワークを可視化することができる．書誌データの抽出には Microsoft Academic Graph の API を利用している．
- Connected Papers [*12] はある論文とつながりが強い数十件の文献をネットワークと合わせて可視化される文献検索サービスである．内部では semanticscholar を利用されている．

本論文の構成

本論文の構成は，まず次章，2 章で，引用構造のネットワーク表現や，ネットワークの性質を捉えるための特徴量を紹介する．その後，対象のネットワーク構造に似た構造を生成するための生成モデルに関連した先行研究を紹介する．その際，すでに提案されたモデルを方針や表現したい特徴などを踏まえて分類してまとめる．また生成モデルと関連のあるリンク予測を取り上げ，問題設定や手法などを比較する．

続いて 3 章では学術論文の引用ネットワークに対する生成モデルを提案する．まずモデル化の対象は著名な Web of Science 書誌データベースから抽出された確率統計分野の引用ネットワークである．このネットワークの特徴を明らかにして，離散時刻上に生成される引用ネットワークの確率的生成モデルを提案する．各文献には粒度が粗い年次の時刻を割り当てられるものとし，各時刻の文献数，引用における引用ノードと被引用ノードの時刻差にもとづく引用率，各文献の参考文献の件数などにもとづき，優先的選択と三角形形成の成長機構にもとづいてネットワークを生成する．提案するモデルは先行研究と比較して，文献の発表時刻について発表順序ではなくデータとして与えられた離散時刻

[*6] https://access.clarivate.com/
[*7] https://arxiv.org/help/api/
[*8] https://scholar.google.co.uk/intl/en/scholar/inclusion.html
[*9] https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/
[*10] https://www.microsoft.com/en-us/research/project/academic-knowledge/
[*11] https://uoa-eresearch.github.io/citations/
[*12] https://www.connectedpapers.com/

を考慮できること，シミュレーション時にデータの利用しないことを条件として検討を行った．先行研究におけるモデルの適応の検証に用いられることが多い典型的な引用ネットワークである，arXivの高エネルギー物理分野から生成した引用ネットワークに対しても提案モデルの有用性を検証した．

4 章では特許文献の引用ネットワークを対象とした生成モデルを提案する．まず特許文献の引用ネットワークはデータに含まれるカテゴリとサブカテゴリにより階層的なクラスタ構造を形成しているため，対象は一部のサブカテゴリに限定した．まず 3 章で提案した学術論文の引用ネットワーク向けの生成モデルを適用したところ，十分な適合が得られないため，各ノードごとの成長機構を切り替えるパラメータを柔軟に設定できるように拡張を実施する．なお拡張したモデルへの検証は，特許の引用ネットワークに加えて，3 章で扱った学術論文の引用ネットワークに対して行った．

5 章で本論文を締めくくる．

第 2 章

引用ネットワークと提案された生成モデル

本章では引用構造のネットワーク表現について定義し，これまでに提案された生成モデルを紹介する．

2.1 引用構造のネットワーク表現

学術論文の引用構造は文献をノードに，引用をエッジに対応させた有向グラフ $G = (V, E)$ で表現できる．ここで文献 i ネットワーク上のノード $v_i \in V$ に，文献 i からの文献 j への引用は有向エッジ $(v_i, v_j) \in E$ にそれぞれ対応する．ここで文献 i が発表された時刻 t_i は $1, 2, \ldots, T$ と正規化されているものとし，各ノード v_i の発表時刻は $\tau(v_i)$ と表記する．

引用ネットワークは一般的に文献はそれよりも古い文献のみを引用の対象とするため，あるエッジ $(v_i, v_j) \in E$ は発表時刻に関して不等式 $\tau(v_i) \geq \tau(v_j)$ を満たす．しかしながら実データにはこの制約を満たさない引用も存在し，短期間で投稿された論文間に引用関係が生じる場合や，また査読プロセスの期間が異なる場合に発生すると考えられる．さらに文献が引用する際に，引用先の文献同士が引用関係をもつことが多い [Holme and Kim, 2002, Wu and Holme, 2009] ことも特徴である．これはノード $v_i \in V$ がノード $v_j, v_k \in V$ と隣接関係 (v_i, v_j) と (v_i, v_k) があるとき，エッジ (v_j, v_k) もしくはエッジ (v_k, v_j) が存在する可能性が高いことに対応する．

図 2.1 や 図 2.2 は引用ネットワークの典型的な形状を表したものである．まず図 2.1 はネットワーク全体を可視化したものだが，図からも伺えるように引用ネットワークは非常に結びつきが強いことが確認できる．さらに図 2.2 はあるノード v_{43845} に着目しその周辺を可視化したものである．ここでノード ID は発表順に割り振られたものとすると，すでに紹介した特徴である，ある文献はそれよりも古いノードのみを引用することが確認できる．またノード v_{43845} と，このノードが引用するノード v_{41752} によって形成される三角形型の引用 $(v_{43845}, v_{41752}, *)$ は 9 件と比較的多いことが確認できる．

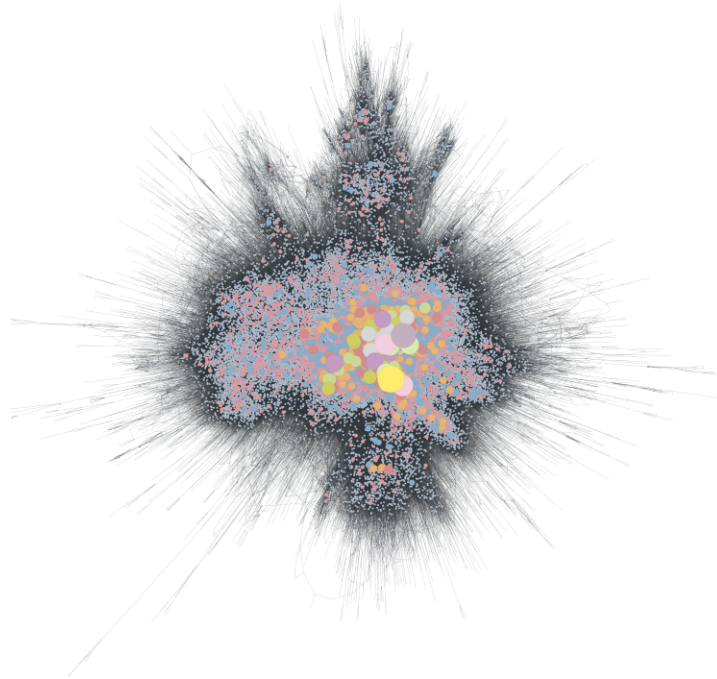


図 2.1. 引用ネットワークの例（全体）

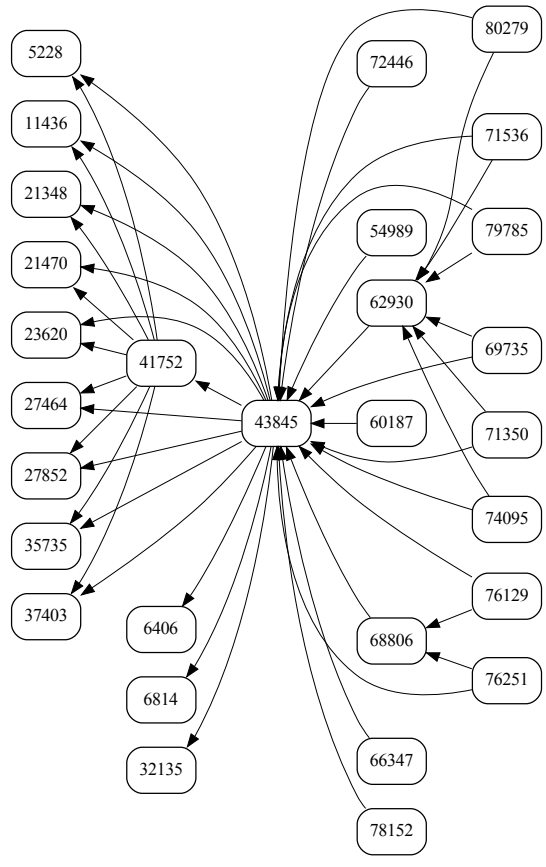


図 2.2. 引用ネットワークの例（ノード v_{43845} に着目）

2.2 ネットワーク特徴量

前節で説明したように，引用構造をネットワークとして表現し，引用構造の比較を対応するグラフ構造の比較として扱うことが可能になった．グラフ構造の類似性を明示的に扱うことは依然として難しいものの，次節で紹介するすでに提案された生成モデルや，次章で提案する我々の生成モデルが，データへ適合しているかを評価する際に必要となる．そこで本研究では，グラフ構造の類似性をグラフ上の特徴量の類似性で近似的し，適合の評価に用いる．本節では，注目するいくつかの特徴量を説明する．

2.2.1 次数分布

まずは典型的な特徴である次数分布について説明を行う．次数分布は 2 ノード間の典型的な特徴量である．$G = (V, E)$ 上のあるノード v の隣接ノード集合を向きに注意して，

$$A_{\mathrm{in}}(v) = \{u \mid (u, v) \in E\} \tag{2.1}$$
$$A_{\mathrm{out}}(v) = \{u \mid (v, u) \in E\} \tag{2.2}$$

と定義する．そして，それらの要素数を向きに注意し，入次数 (in-degree) $d_{\mathrm{in}}(v) = |A_{\mathrm{in}}(v)|$ と，出次数 (out-degree) を $d_{\mathrm{out}}(v) = |A_{\mathrm{out}}(v)|$ で定義する．ここで $|\cdot|$ は要素数を表すものとする．

さらに入次数分布 (in-degree distribution) と出次数分布 (in-degree distribution) はそれぞれ

$$p_{\mathrm{in}}(k) = \frac{|\{v \mid v \in V,\ d_{\mathrm{in}}(v) = k\}|}{|V|} \tag{2.3}$$
$$p_{\mathrm{out}}(k) = \frac{|\{v \mid v \in V,\ d_{\mathrm{out}}(v) = k\}|}{|V|} \tag{2.4}$$

と，次数 k に対応するノードの割合で定義される．引用ネットワークにおける，対象の文献が引用される件数は入次数で，対象の文献が引用する件数は出次数に対応するため，自然な解釈が可能である．

2.2.2 ノードが参加する三角形数の分布

ノードの周辺に存在する三角形数の分布 (Node triangle participation) [Holme and Kim, 2002, Wu and Holme, 2009, Leskovec et al., 2010] について説明を進める．三角形数の分布は典型的な 3 ノード間の特徴量となる．あるノード $v \in V$ が参加する三角形の数は

$$\delta(v) = |\{(v, v_1, v_2) \mid v_1, v_2 \in A(v),\ (v_1 \in A(v_2) \text{ or } v_2 \in A(v_1))\}| \tag{2.5}$$

と定義できる．$\delta(v)$ はローカルクラスタリング係数 (local clustering coefficient) [Watts and Strogatz, 1998] における，ノード v で形成されうる最大の三角形数で正規化する前の値（件数）と捉えることができる．$\delta(v)$ は文献 v に隣接するノードと形成される，3 ノードの組合せ (v, v_1, v_2) の件数に対応している．この定義では $A_{\mathrm{in}}(v)$ と $A_{\mathrm{out}}(v)$ を区別せずに含めているが，それぞれ被引用

と引用に対応することになるため，書誌学では明確に区別されることも多い．本研究では計算の簡便さを優先して向きを考慮しない件数を用いる．三角形数の分布は次数分布と同様に三角形の数 k に対応するノードの割合で定義される．

$$p_{\mathrm{tri}}(k) = \frac{|\{v \mid v \in V,\ \delta(v) = k\}|}{|V|} \tag{2.6}$$

図 2.3 はあるノードの周辺に存在する三角形型のエッジの密集構造の例であり，ノード v は 3 件の三角形をもつ $(\delta(v) = 3)$.

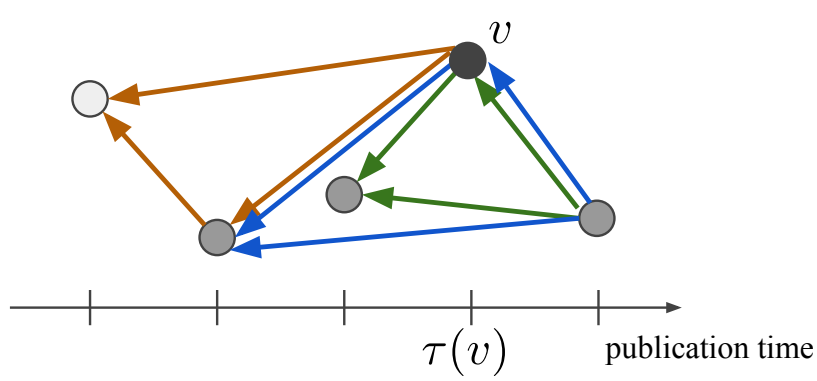


図 2.3. 引用ネットワーク上の三角形型の引用構造の例

2.2.3 グラフ隣接行列に対する Scree plot

対象のグラフ $G = (V, E)$ を隣接行列 (adjacency matrix) $A_G = \{0, 1\}^{|V| \times |V|}$ で表現する．隣接行列 A_G の各要素 (i, j)-成分は，対象のグラフ G にエッジ (v_i, v_j) が存在するときに 1 を，そうではないときに 0 となる．scree plot は隣接行列 A_G に対する特異値分解 $A_G = U_G \cdot \Sigma_G \cdot V_G$ により得られた対角行列 Σ_G の各成分である特異値 [Farkas et al., 2001] を大きい順に並べた $(\sigma_1, \sigma_2, ...,)$ を可視化したものである．先行研究 [Leskovec et al., 2010] では上位 200 要素を可視化している．scree plot はネットワークに広域にわたる特徴を表現していると考えられる．

2.2.4 その他の指標

中心性指標は典型的なネットワーク特徴量であり，中心的となる重要なノードやエッジの選択に役立つ．前述の出次数や入次数は合わせて，次数中心性として捉えられることも多い．最短パスを用いた中心性指標として近接中心性 (closeness centrality) [Sabidussi, 1966] や媒介中心性 (Betweenness Centrality) [Freeman, 1977] が挙げられる．特に媒介中心性には効率的なアルゴリズムとして [Brandes, 2001] が存在し，効率的な実装 [Yasui et al., 2011] [*1]などが利用可能である．また媒介中心性には様々な亜種が存在する [Brandes, 2008].

[*1] `https://bitbucket.org/yuichiro_yasui/netal`

その他，固有ベクトル中心性 (Eigenvector centrality) [Bonacich, 1987] は対象のグラフを表現した隣接行列の最大固有値に対応する固有ベクトルを用いた指標である．著名な PageRank [Brin and Page, 1998] は固有ベクトル中心性の亜種となる．

ネットワーク・モチーフ (Network motifs) は対象のネットワークに含まれる小さな接続パターンである．次数（2 ノード間の接続）や，三角形（3 ノード間の接続）もモチーフの 1 種として捉えることができる．効率的に数え上げるアルゴリズムとして [Wernicke, 2006] などが利用可能である．

Hop plot は最短パス長と，それに該当するノードのペア数の累積分布である [Leskovec et al., 2010]．引用ネットワークと関連付けた解釈は簡単ではないものの，ネットワークの形状を示す典型的な特徴量である．なお引用ネットワークでは直径と呼ばれる，最も遠い 2 ノード間の最短パス長は，比較的小さい．

2.3 一般的なネットワーク生成モデル

2.3.1 Erdös–Rényi モデル

Erdös–Rényi モデル [Erdös and Rényi, 1959] はノード数 n に対してエッジを確率 p で生成する．つまりすべてのノード対に対して確率 $0 < p < 1$ でエッジを生成し，確率 $1 - p$ で生成しない．このモデルはネットワーク生成モデルとして最初に提案されたものであり，種々の数学的性質が証明されている [Erdos and Renyi, 1960]. 主に n や p を調整したときの連結成分に関する性質を取り扱っている．あるノードは他の $n - 1$ ノードと確率 p で隣接するため，次数分布 p_k は二項分布

$$p_k = \binom{n-1}{k} p^k (1-p)^{n-1-k} \tag{2.7}$$

に従う [Newman et al., 2001]．平均次数は $(n-1)p$ である．さらに n が十分に大きいとき，ポアソン分布

$$p_k \simeq \frac{(np)^k e^{-np}}{k!} \tag{2.8}$$

に収束する．ランダムグラフと呼ばれ，パス長が短いなどの特徴を有する．しかしながらソーシャル・ネットワークやインターネットの構造はポアソン分布に従う次数分布とは大きく異なる次数分布を持つことが指摘されている [Watts and Strogatz, 1998, Newman et al., 2001].

2.3.2 Watts–Strogatz モデル

Watts–Strogatz モデル [Watts and Strogatz, 1998] はまず n ノードをリング上に配置し，最も近い k (k が奇数の場合は $k-1$) の隣接ノードへのエッジを生成する．その後，各エッジ (u, v) に対して，確率 $0 \leq p \leq 1$ で新しいエッジ (u, w) に置き直す．ここでノード w は一様にランダムに選択されるものとする．ランダムグラフに対してパス長が短い性質を保持しつつ，クラスタ性を調整できる．確率 p を 0 から 1 へ調整することで，リング上のネットワークから，スモールワールド性をもつネットワーク，ランダム・ネットワークへと変化する．その際の次数分布は，0 のときは全ノードで同じ，1 のときはほぼランダムネットワークとなるためポアソン分布に従う．文献 [Watts and Strogatz, 1998] 内で，スモールワールド・ネットワークである線虫の神経網ネットワーク，送電網ネットワーク，映画俳優の共演ネットワークなどへの適合が示されている．ここでスモールワールド性とは平均パス長が L とノード数 n に対して $L \propto \log n$ が成り立つことであり，そのような性質をもつネットワークをスモールワールド・ネットワークと呼ぶ.

2.3.3 *dk*-series モデル

dK-series モデル [Mahadevan et al., 2006], [Orsini et al., 2015] はグラフ上の d ノードからなる連結な部分グラフの d 次元の次数相関を同時に考慮する．ここで $0K$ $(d = 0)$ は平均次数 $\bar{k}$ が与え

られるため Erdös–Rényi モデルと一致する．$1K$ $(d = 1)$ は与えられた次数分布を，$2K$ $(d = 2)$ は 2 ノード間の次数分布を同時分布をそれぞれ考慮することができる．さらに $3K$ $(d = 3)$ では様々なネットワーク上の特徴を表現でき，例えば媒介中心性などの大域的な特徴も表現できると報告されている．しかしながら $d > 2$ に対しては効率的な推定方法が見つかっておらず実ネットワークへの適用が容易ではない．そのため 2 ノード間の次数分布に加えて平均クラスタリング係数を考慮する $2.1K$ や, 次数に依存する平均クラスタリング係数を考慮する $2.5K$ など， $2K$ をベースに拡張しつつ実用的なモデルが提案されている．$2K$ や $2.5K$ に関する実装 [*2]が利用可能である．

2.3.4 Stochastic Kronecker Graph モデル

Stochastic Kronecker Graph (SKG) モデル [Leskovec et al., 2010] は各要素の取り得る範囲が $0 \leq \Theta_{ij} \leq 1$ となる N_1-次正方行列 Θ に対して，クロネッカー積を $k-1$ 回適用した $\Theta^{[k]}$ の各 (i, j)–成分をノード v_i からノード v_j へのエッジ (v_i, v_j) の生成確率とする生成モデルである．ここで $\Theta^{[k]}$ は N_1^k-次正方行列となり，ノード数が N_1^k，エッジ数の期待値が $\left(\sum_{i,j=1}^{N_1} \Theta_{ij}\right)^k$ となるネットワークが生成される．行列 Θ の次元 N_1 は 2 (つまり 2×2) でも十分に様々なネットワーク構造に適合すると報告されている．なお全ての要素が p となる行列 $\Theta = \begin{pmatrix} p & p \\ p & p \end{pmatrix}$ ($N_1 = 2$ の場合) を初期行列に用いると Erdös–Rényi モデルに対応する．生成確率の決定に複数回のクロネッカー積を用いるため，図 2.4 のように初期行列に応じた自己相似構造をもつ．

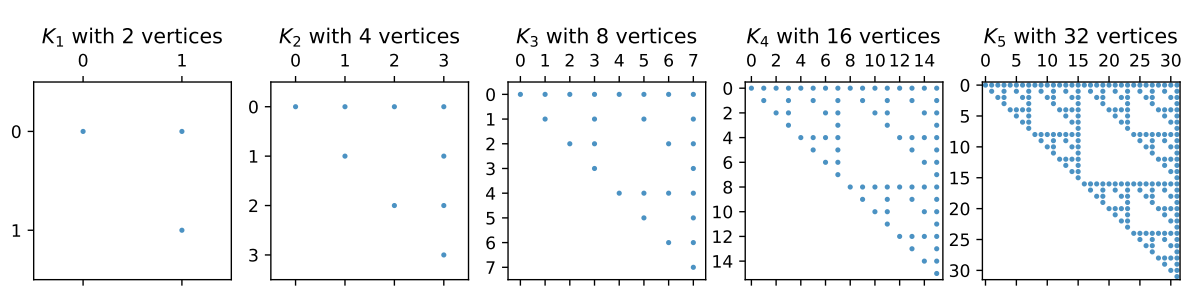


図 2.4. Kronecker graph モデルにおける自己相似構造

SKG におけるエッジの生成確率を独立した二項分布に従う確率変数としてモデル化できるため，パラメータ Θ の推定は尤度関数

$$P(G \mid \Theta^{[k]}, \sigma) = \prod_{(u,v) \in G} \Theta^{[k]}[\sigma_u, \sigma_v] \prod_{(u,v) \notin G} (1 - \Theta^{[k]}[\sigma_u, \sigma_v]) \tag{2.9}$$

からなる最適化問題 $\arg\max_\Theta P(G \mid \Theta^{[k]})$ により最尤推定値 $\hat{\Theta}$ を得ることができる．ここで σ はノードの順列を表し，i 番目の要素は σ_i に対応する．このように最適化問題はノードの順序 ($N!$ 通り) とエッジの候補 (N^2 通り) を同時に考えるため，計算量 $\mathcal{O}(N!N^2)$ と大きい．エッジ数 M に比例する計算量 $\mathcal{O}(M)$ へ削減できる Metropolis sampling にもとづく推定手法を提案している．

[*2] http://www.minasgjoka.com/2.5K/instructions/

SKG モデルのシミュレーションでは推定されたパラメータ $\hat{\Theta}$ に対し，目的のノード数になるように必要な回数のクロネッカー積を適用して生成確率を生成する．各エッジを独立に生成するかどうかを評価する場合は $\mathcal{O}(N^2)$ が必要になるものの，エッジ数 M に比例する計算量 $\mathcal{O}(M)$ に削減する近似手法を提案している．SKG モデルは SNAP パッケージ [Leskovec and Sosič, 2016] にパラメータ推定のため関数 `kronfit`，シミュレーションのための関数 `krongen` が用意されている．

なお Kronecker graph は高性能計算 (High-performance computing) 分野における Graph500 ベンチマークや Green Graph500 ベンチマーク [*3] での計算対象に設定されている．これらのベンチマークでは高性能な計算機上でのグラフ探索（幅優先探索や単一始点最短パス）に対する効率的なアルゴリズムや高速な実装について議論されている [Yasui et al., 2013].

2.4 引用ネットワークに対する成長モデル

引用ネットワークのための生成モデルは主にネットワークの成長モデルに分類できる．成長モデルは，最小限の連結成分として与えられたネットワーク初期状態に対して，逐次的にノードやエッジを追加することで，ネットワークを成長させる．次章で比較対象となる Barabási–Albert モデル [Barabási and Albert, 1999]，Holme–Kim モデル [Holme and Kim, 2002]，Wu–Holme モデル [Wu and Holme, 2009] はいずれも成長モデルに分類することができる．

2.4.1 Barabási–Albert モデル

Barabási–Albert モデル [Barabási and Albert, 1999] は 1 ノードずつを生成し，そのノードから k エッジを生成するという**成長機構**を n 回繰り返して，ネットワークを成長させる．

追加されたノード v_i から，すでにネットワーク上に存在するノード v_j へエッジを生成する確率はノード v_j の重要度に比例した確率 $\Pi_{\mathrm{PA}}(v_j)$ で設定される．確率的に v_j を選択することでエッジ (v_i, v_j) をネットワークに追加する．この機構は**優先的選択** (preferential attachment; PA) と呼ばれる．Barabási–Albert モデルはノード v_j の重要度をその入次数 $d_{\mathrm{in}}(v_j)$ で近似するため，選択確率は $\Pi_{\mathrm{PA}}(v_j) \sim d_{\mathrm{in}}(v_j)$ となる．

生成されるネットワークはスモールワールド性やスケールフリー性をもつことが示されている．ここでスケールフリー性とは次数分布はべき乗則 (power law) $p(k) \sim k^{\gamma}$ に従うことで，Barabási–Albert モデルから生成されるネットワークは $\gamma = -3$ となる．実ネットワークにおいてもべき乗則に従う次数分布をもつことが知られており，映画俳優の共演ネットワークは $\gamma_{\mathrm{actor}} = -2.3 \pm 0.1$，インターネットのリンク構造は $\gamma_{\mathrm{www}} = -2.1 \pm 0.1$，送電網ネットワークは $\gamma_{\mathrm{power}} = -4$ となる [Barabási and Albert, 1999].

引用ネットワークに対する優先的選択機構は Price [Price, 1976] が, Barabási と Albert [Barabási and Albert, 1999] よりも先に提案している．

このモデルを用いた，シミュレーションのためのアルゴリズムは付録の Algorithm 5 にまとめる．

[*3] `https://graph500.org/`

2.4.2 Holme–Kim モデル

引用ネットワークは前述の PA 機構だけで構築されるネットワークに比べて，三角形の引用構造が多く形成される．これはある文献に着目したとき，引用する文献や引用される文献同士での引用が多く形成されることである．

Holme–Kim モデル [Holme and Kim, 2002] は Barabási–Albert モデルと同様に 1 ノードずつを生成し，そのノードから k エッジを生成するという成長機構を n 回繰り返して，ネットワークを成長させる．ノード v_i が追加されたとき，PA 機構か**三角形形成** (Triad formation; TF) 機構のどちらかでノード v を選択し，エッジ (v_i, v) を生成する．確率 β で TF 機構が実行され，$1-\beta$ で PA 機構が実行される．

PA 機構はすでにネットワーク上に存在するノード v_j を確率 $\Pi_{\mathrm{PA}}(v_j)$ で選択し，エッジ (v_i, v_j) をネットワークに追加する．Barabási–Albert モデルと同様に $\Pi_{\mathrm{PA}}(v_j) \sim d_{\mathrm{in}}(v_j)$ を用いる．一方，TF 機構は直前の PA 機構で選択された v_j の隣接ノード集合からノード $v_k \in A(v_j) \setminus \{v_i\}$ を $\Pi_{\mathrm{TF}}(v_k)$ に比例する確率で 1 つ選択し，エッジ (v_i, v_k) をネットワークに追加する．TF 機構により少なくとも 1 つの $\{v_i, v_j, v_k\}$ からなる三角形が形成される．$\Pi_{\mathrm{TF}}(v_j) \sim 1$ に設定される．

次数分布は Barabási–Albert と同様にべき乗則に従い $p(k) \sim k^{-3}$ となる．Barabási–Albert と比較し大きいクラスタ係数を取ることができる．このように Holme–Kim モデルはスモールワールドかつスケールフリーのネットワークを生成できる．シミュレーションのためのアルゴリズムは付録の Algorithm 6 にまとめる．

2.4.3 コピーにもとづく生成モデル

Krapivsky と Redner [Krapivsky and Redner, 2005] は引用ネットワークにおける文献引用の大部分が引用した文献の参考文献を単にコピーしただけであると指摘し，コピーをもとにした成長モデルを構築した．このモデルはノード v_i が追加されたとき，すでにネットワーク上に存在するノード集合からランダムに 1 つノード v_j を選択しエッジ (v_i, v_j) を生成する．その後，v_j が引用するノード集合に対するエッジ $\{(v_i, v_k) \mid v_k \in A_{\mathrm{out}}(v_j)\}$ をネットワークに追加する．

2.4.4 Wu–Holme モデル

Wu–Holme モデル [Wu and Holme, 2009] は Holme–Kim モデルをもとに，時刻に応じてエッジの引用確率が変化する**経時変化**に着目した．成長機構は添え字順にノード v_i, $i \in \{1, 2, ..., n-1\}$ を 1 つずつネットワークに生成する．ノード v_i が追加されたとき，PA 機構か TF 機構のどちらかでノード v を選択し，エッジ (v_i, v) を生成する．ノード v_i が生成するエッジ数は k_i となる．確率 β で TF 機構が実行され，$1-\beta$ で PA 機構が実行される．

PA 機構はすでにネットワーク上に存在するノード v_j を確率 $\Pi_{\mathrm{PA}}(v_j)$ で選択し，エッジ (v_i, v_j) をネットワークに追加する．ここでノードの添え字に注意し $i > j$ のときにノード v_i からノード v_j

へ引用年齢を $i-j$ で設定する．PA 機構における各ノード v_j の選択確率は（次数ではなく）引用年齢を用いた $\Pi_{\mathrm{PA}}(v_i, v_j) \sim (i-j)^{\alpha}$ で設定する．なお TF 機構については Holme–Kim モデルと同じものを用いる．引用ネットワークにおいて α は -1 などの負値で推定され，時刻差が大きくなると引用の割合が小さくなるという性質を表現している．

Wu–Holme モデルで生成するネットワークに対し，ローカル・クラスタ係数やグローバル・クラスタ係数などの理論的な上界・下界については文献 [Oliveira et al., 2018] で示されている．

シミュレーションのためのアルゴリズムは付録の Algorithm 7 にまとめる．

2.4.5 Chang–Phoa–Nakano モデル

Chang–Phoa–Nakano モデル [Chang et al., 2021] は重要度と時刻差による引用率を考慮して, 離散時刻を扱った生成モデルである．シグモイド関数に従う各時刻の文献数, tapered Pareto distribution に従う入次数で近似する重要度，ガンマ分布の確率密度関数の定数倍に従う時刻差ごとの引用率を用いてモデル化を行った．シミュレーションにより，各ノードにおいて距離がちょうど 2 となるエッジ数に対応する，距離 2–入次数への適合を示した．

2.5 引用ネットワークに対するモデル適応の確認

図 2.5 は arXiv 書誌データより生成されたネットワーク `arXiv-HepTh` に対する生成モデルの適合の度合いを比較したものである．このネットワークは様々な生成モデルでのモデル適応の検証に使用されている．引用ネットワークの構築の条件などの詳細は次章 3 章で述べる．

まずデータ (Real network) に対して，Erdös–Rényi (ER) モデルはいずれのネットワーク特徴量も適応が十分でない．続いて成長モデル Barabási–Albert (BA), Holme–Kim (HK), Wu–Holme (WH) はいずれも優先的選択 (PA) 機構により入次数に適合している．さらに Wu–Holme では三角形形成 (TF) 機構と経時変化を組み合わせたことにより，三角形数の分布に対しても適合していることが確認できる．また Stochastic Kronecker Graph (SKG) は出次数分布への適応を確認できたものの，それ以外で適応が十分でない．SKG の初期行列は文献 [Leskovec et al., 2010] で得られた推定結果 $\begin{pmatrix} 0.990 & 0.440 \\ 0.347 & 0.538 \end{pmatrix}$ を用いた．

これらの結果から，成長モデルは引用ネットワークの性質を表現できており，次章で提案するモデル適合を検証する際に Barabási–Albert (BA), Holme–Kim (HK), Wu–Holme (WH) を比較モデルとして用いる．

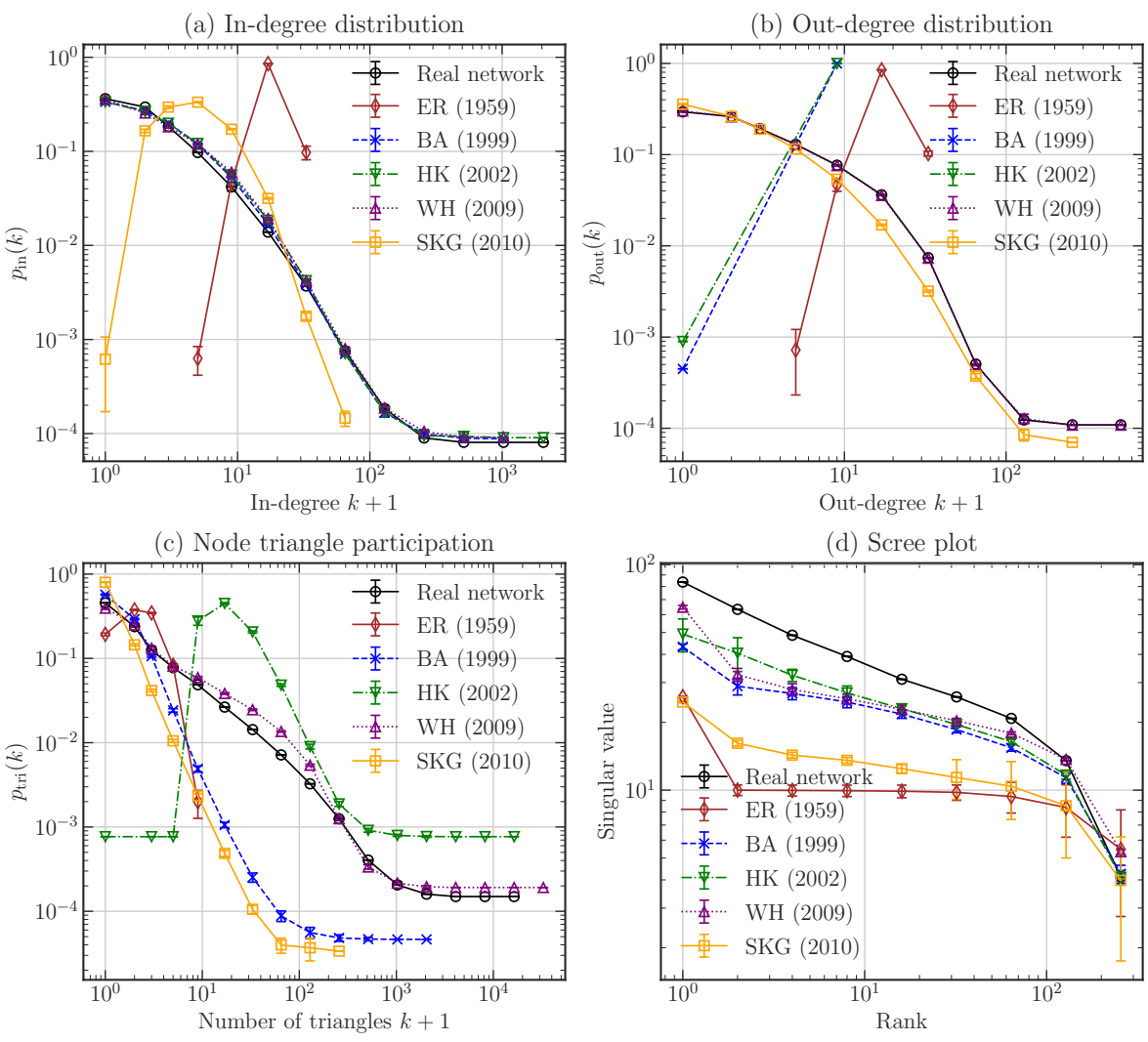


図 2.5. 引用ネットワーク `arXiv-HepTh` に対するモデル適合の比較

2.6 生成モデルとリンク予測

生成モデルと関連があるリンク予測 (Link prediction) について説明を行う．リンク予測はある時点のネットワーク構造を既知として，その後ある程度近い将来に生じる可能性が高いエッジを予測する．リンク予測はネットワーク生成モデルにおける，追加されたノードから既存のノードへエッジを生成する部分に注目した問題設定とみなすこともできる．

リンク予測は主に無向グラフ $G = (V, E)$ を入力として，生成されうるエッジの可能性であるノード対 $U \in V \times V$ のうち既知のエッジ集合を除外した $E_p = U - E$ を対象に，エッジの候補 $(u, v) \in E_p$ がされるかどうか予測する．

最も典型的なアプローチはエッジの候補 $(u, v) \in E_p$ に対して存在の確からしさを表す指標を構成する手法である [Liben-Nowell and Kleinberg, 2003, Linyuan and Zhou, 2011]．表 2.1 には，生成モデルに関連が強い指標をまとめる．$\mathrm{score}_{\mathrm{PA}}$ は優先的選択そのものであり，$\mathrm{score}_{\mathrm{CN}}$, $\mathrm{score}_{\mathrm{Jaccard}}$, $\mathrm{score}_{\mathrm{AA}}$, $\mathrm{score}_{\mathrm{RA}}$ はいずれも三角形形成のように 3 ノード間（三角形）の特徴に着目している．

表 2.1. リンク予測に用いるエッジ指標

指標	定義
Common neighbors (CN)	$\mathrm{score}_{\mathrm{CN}}(u, v) = \lvert A(u) \cap A(v) \rvert$
Jaccard index	$\mathrm{score}_{\mathrm{Jaccard}}(u, v) = \dfrac{\lvert A(u) \cap A(v) \rvert}{\lvert A(u) \cup A(v) \rvert}$
Preferential Attachment Index (PA)	$\mathrm{score}_{\mathrm{PA}}(u, v) = \lvert A(u) \rvert \cdot \lvert A(v) \rvert$
Adamic–Adar Index (AA)	$\mathrm{score}_{\mathrm{AA}}(u, v) = \displaystyle\sum_{z \in A(u) \cap A(v)} \dfrac{1}{\log \lvert A(z) \rvert}$
Resource Allocation Index (RA)	$\mathrm{score}_{\mathrm{RA}}(u, v) = \displaystyle\sum_{z \in A(u) \cap A(v)} \dfrac{1}{\lvert A(z) \rvert}$

これらの指標は生成モデルのエッジ生成と関連が高いものの，リンク予測は生成モデルとは異なる点も存在する．まず生成モデルにおいてリンク予測の問題設定と類似性が高い，ノード u が追加されノード v へのエッジを生成する段階を考える．生成モデルでは常にネットワーク上に存在しない新たなノードを追加するため，追加された直後のノード u は常に隣接ノードが存在しない．そのため上記の指標はいずれも 0 となり機能しない．リンク予測が十分に機能するためには，リンク候補のノード対 (u, v) に対しても十分に隣接エッジが存在する必要がある．

なお表 2.1 にまとめたアプローチ以外にも，最短パスやランダムウォークを用いた指標，グラフの隣接行列表現に対する特異値分解による行列分解の結果を機械学習に用いる特徴量とするアプローチなども存在する [Liben-Nowell and Kleinberg, 2003, Linyuan and Zhou, 2011]．また近年ではグラフ畳み込みネットワークやグラフニューラルネットワークに関連するアプローチも盛んに研究されている [Zhang and Chen, 2018].

第 3 章

学術論文の引用ネットワークに対する確率生成モデル

本章ではまず対象の引用ネットワークの構築方法について説明し，時刻を考慮した特徴量とそれにもとづく生成モデル [Yasui and Nakano, 2021, Yasui and Nakano, 2022b] を提案する．シミュレーションによりモデルの適合を示す.

3.1 引用ネットワーク WoS-Stat の構築

Web of Science (WoS) [Clarivate Analytics, 1997] は著名な大規模書誌データベースであり，データベースには各レコード「タイトル, 著者情報, 発表時刻, アブストラクト, 発表されたジャーナル, ジャーナルに紐付く事前定義されたカテゴリ, 参考文献のリスト」などが格納されている．WoS 全体となる 1981–2016 年の書誌データでは 2.095 億件の文献と 10.61 億件の引用と非常に大きいため，カテゴリ “Statistics and Probability” に紐付くジャーナルから発表された文献間の引用のみに着目した引用ネットワーク WoS-Stat をモデリングの対象とする.

統計数理研究所 URA チームにより，WoS 書誌データはグラフデータベース Neo4j に格納されている．このグラフベースに対して，Neo4j 用のクエリ言語 Cypher により抽出・集計することができる．図 3.1 は Neo4j 上のデータモデルと呼ばれるグラフ構造である.

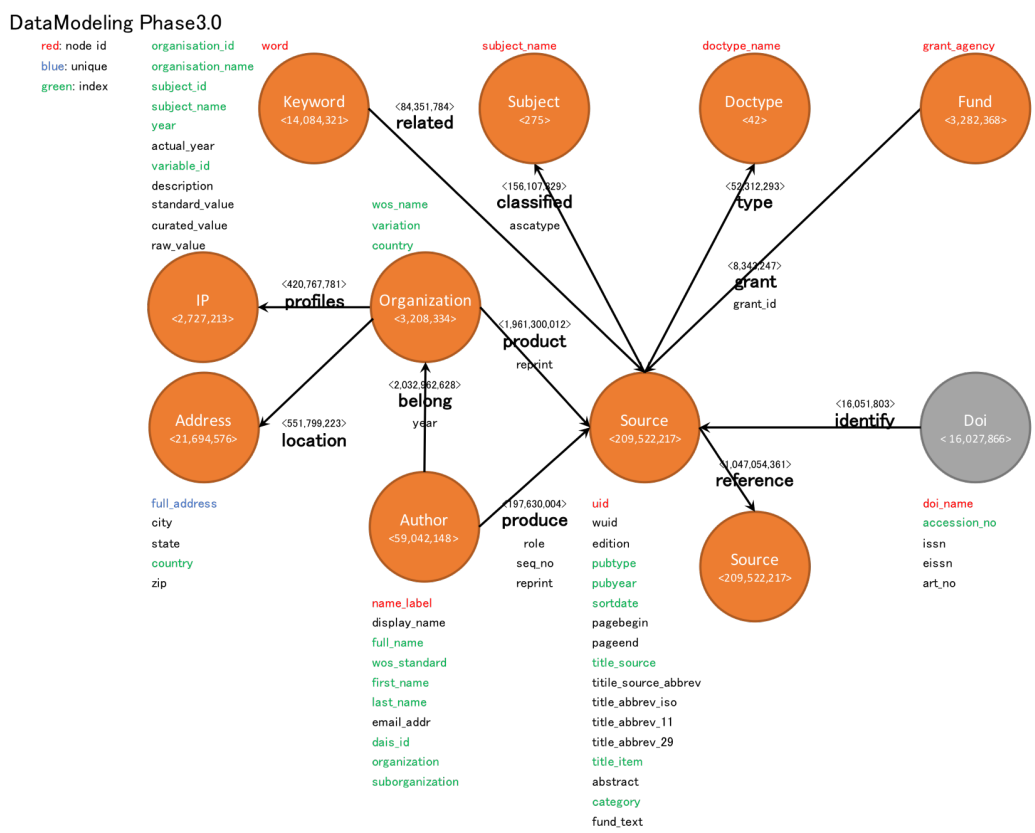


図 3.1. WoS 書誌データが格納された Neo4j のデータモデル

図 3.2 はデータベースからの抽出対象となり，“Statistics and Probability” に紐付くジャーナルから発表された文献と，その文献間の引用構造となる．なお各文献ノードには論文 ID となる `uid` と発表時刻を表す `pubyear` が付与されている．Listing 3.1 は統計数理研究所 URA チームが運営する Neo4j から `WoS-Stat` を抽出するための Cypher クエリである．名称が `"Statistics & Probability"` となるカテゴリ `si` と `sj` にそれぞれ紐付く文献 `vi` と `vj` の引用構造を抽出対象としている．表 3.1 に Listing 3.1 で示した Cypher クエリの実行結果のうち最初の 5 レコードを示す．

表 3.1. Neo4j からの抽出された 1,106,622 レコードの最初の 5 レコード

vi.pubyear	vi.sortdate	vi.uid	vj.pubyear	vj.sortdate	vj.uid
2016	2016-07-01	WOS:000381591300020	2016	2016-06-01	WOS:000374563100009
2016	2016-11-01	WOS:000381839500002	2016	2016-04-01	WOS:000374235800012
2016	2016-01-01	WOS:000379257000001	2016	2016-01-01	WOS:000379257000009
2016	2016-12-01	WOS:000390007100009	2016	2016-06-01	WOS:000390006800002
2016	2016-01-01	WOS:000374951900003	2016	2016-01-02	WOS:000364327900001

本研究で用いた `WoS-Stat` は Dryad データレポジトリ [Yasui and Nakano, 2022a] で公開されている．データの利用方法については付録 B.1 にまとめる．

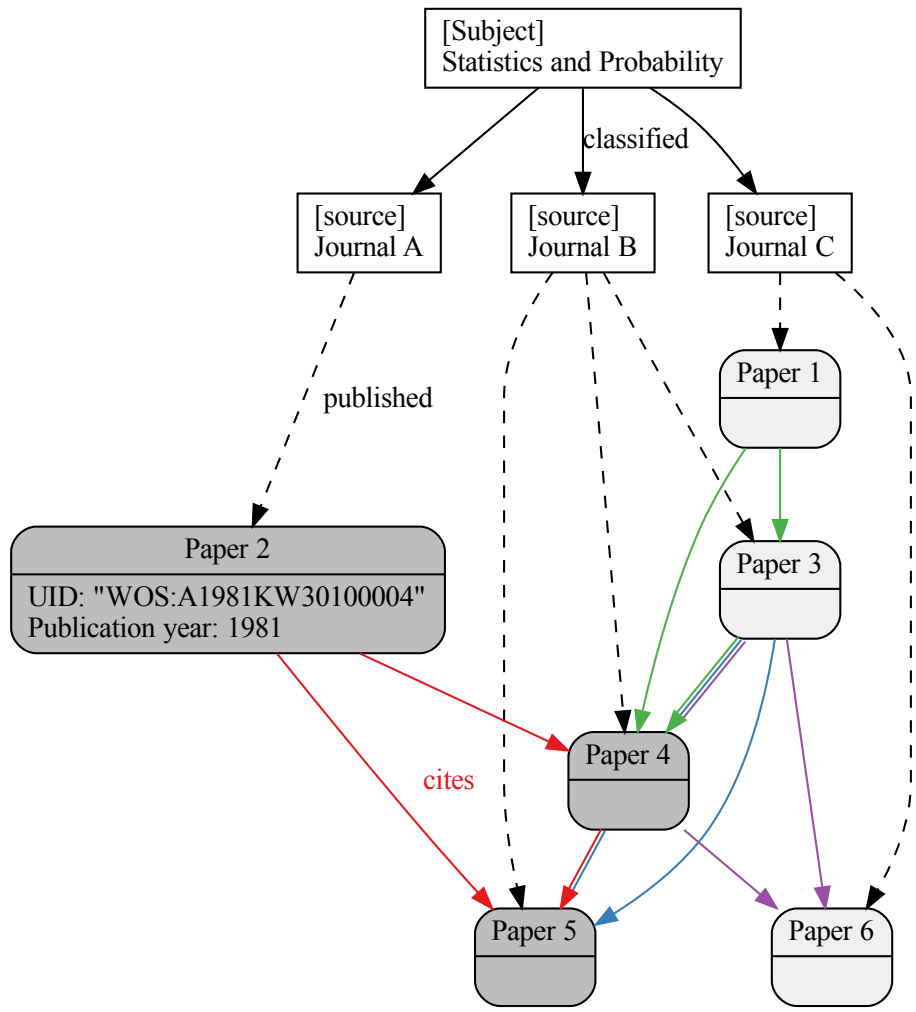


図 3.2. WoS 書誌データが格納された Neo4j 上での抽出対象である確率統計分野の文献とその引用

Listing 3.1. WoS 書誌データが格納された Neo4j からのデータ抽出に用いた Cypher クエリ

```
MATCH
    (si:Subject)<-[:classified]-(vi:Source)
MATCH
    (sj:Subject)<-[:classified]-(vj:Source)
MATCH
    (vi:Source)-[r:reference]->(vj:Source)
WHERE
        si.subject_name = "Statistics␣&␣Probability"
    and sj.subject_name = "Statistics␣&␣Probability"
RETURN
    distinct vi.uid, vi.pubyear, vi.sortdate, vj.uid, vj.pubyear, vj.sortdate
;
```

3.2 WoS-Stat の基本的な性質

WoS-Stat の基本的な性質についてまとめていく．WoS-Stat の文献の発表時刻の範囲は 1981 年から 2016 年までの 36 年間となり，179,483 文献と 1,106,622 引用からなる引用ネットワークとなった．WoS-Stat には 6,411 件の書籍が含むものの，本研究における事前の検証によってネットワーク特徴量への影響は小さく分析上問題ないと判断した．なお “Statistics and Probability” カテゴリに紐付いたジャーナルは “Mathematics” や “Computer Science” などのカテゴリにも紐付いている．表 3.2 は WoS-Stat における文献数が上位 10 件の学術雑誌（ジャーナル）である．

表 3.2. WoS-Stat における文献数が上位 10 件の学術雑誌（ジャーナル）

No.	学術雑誌（ジャーナル）名	文献数
1	BIOINFORMATICS	9,268
2	COMMUNICATIONS IN STATISTICS-THEORY AND METHODS	7,559
3	STATISTICS IN MEDICINE	7,338
4	STATISTICS & PROBABILITY LETTERS	6,857
5	FUZZY SETS AND SYSTEMS	6,705
6	JOURNAL OF STATISTICAL PLANNING AND INFERENCE	5,790
7	JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION	5,045
8	COMPUTATIONAL STATISTICS & DATA ANALYSIS	4,719
9	BIOMETRICS	4,707
10	ANNALS OF STATISTICS	4,069

以後の分析では発表時刻として発表年を用いる．これは書誌データの発表時刻は日次フォーマットであるものの，発表年や発表月の精度となる文献が多く含むためである．図 3.3 は WoS-Stat における発表年ごとの文献数を表したものである．1980–2010 年の範囲で増大し，2010 年以降は増加は鈍化している．

図 3.4 は WoS-Stat における時刻 t_i の文献から時刻 t_j の文献への引用数を (t_i, t_j)-成分とした行列とそのヒートマップである．t_i と t_j には $1, 2, .., T$ と正規化し時刻を用いる．基本的に $t_i \geq t_j$ となることや，例外の存在を確認できる．また時刻が $t_i \leq 30$ 以降に着目すると，時刻差が小さいときは引用の割合が大きく，時刻差が大きくなると割合は小さくなる，しかしながら同じ時刻内での引用は少ないといった性質が観察できる．

図 3.5 は WoS-Stat 上の (a) 入次数分布 p_{in} と (b) 出次数分布 p_{out}，(c) 三角形数の分布 p_{tri}，(d) scree plot を表している．図 (a), (b), (c) は両対数で描画していること，また $k = 0$ をプロットするため横軸は $+1$ となる点に注意されたい．(a), (b), (c) はいずれも裾の重い分布であることを確認できる．また (b) において，10.2% の文献が出次数をもたない理由については，同一分野内に引用が存在しないことや，文献が引用したのはデータの対象外となる 1981 年以前の文献への引用のみであっ

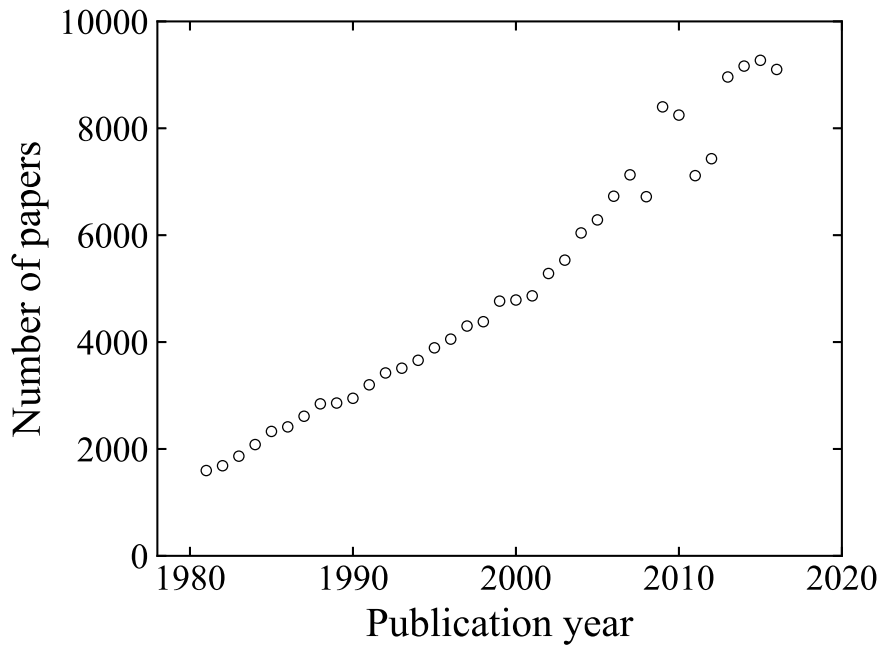


図 3.3. WoS-Stat における時刻ごとの文献数の推移

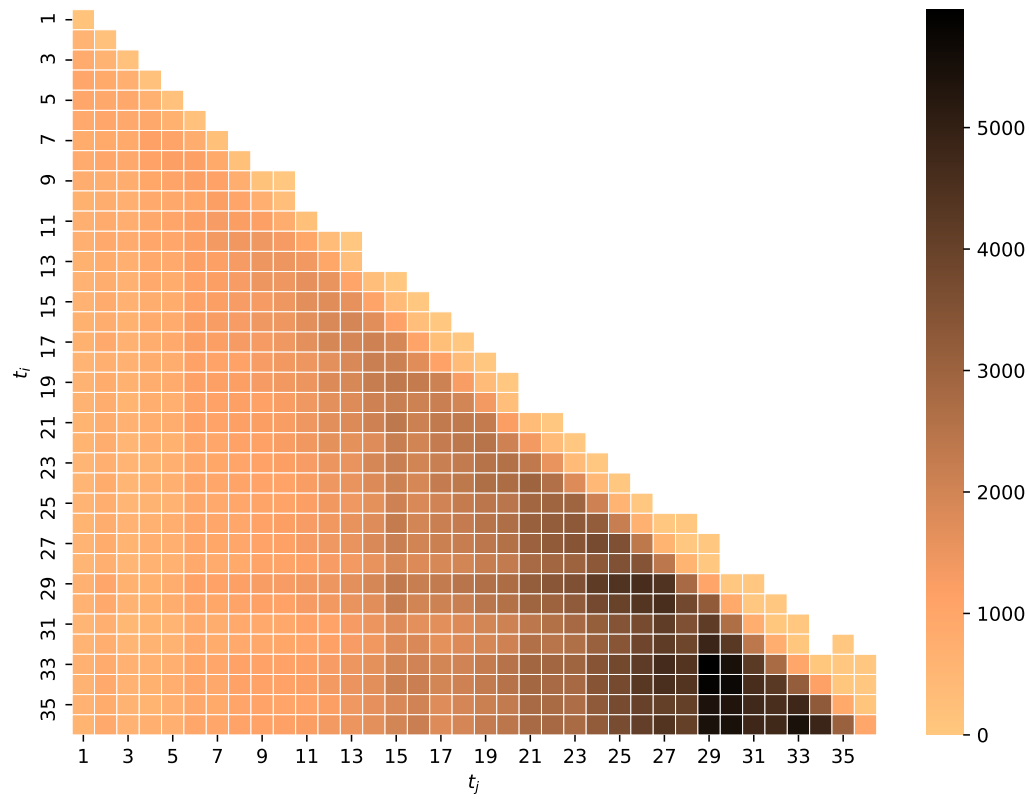


図 3.4. ある時刻 t_i からある時刻 t_j までの引用数

た場合などが考えられる.

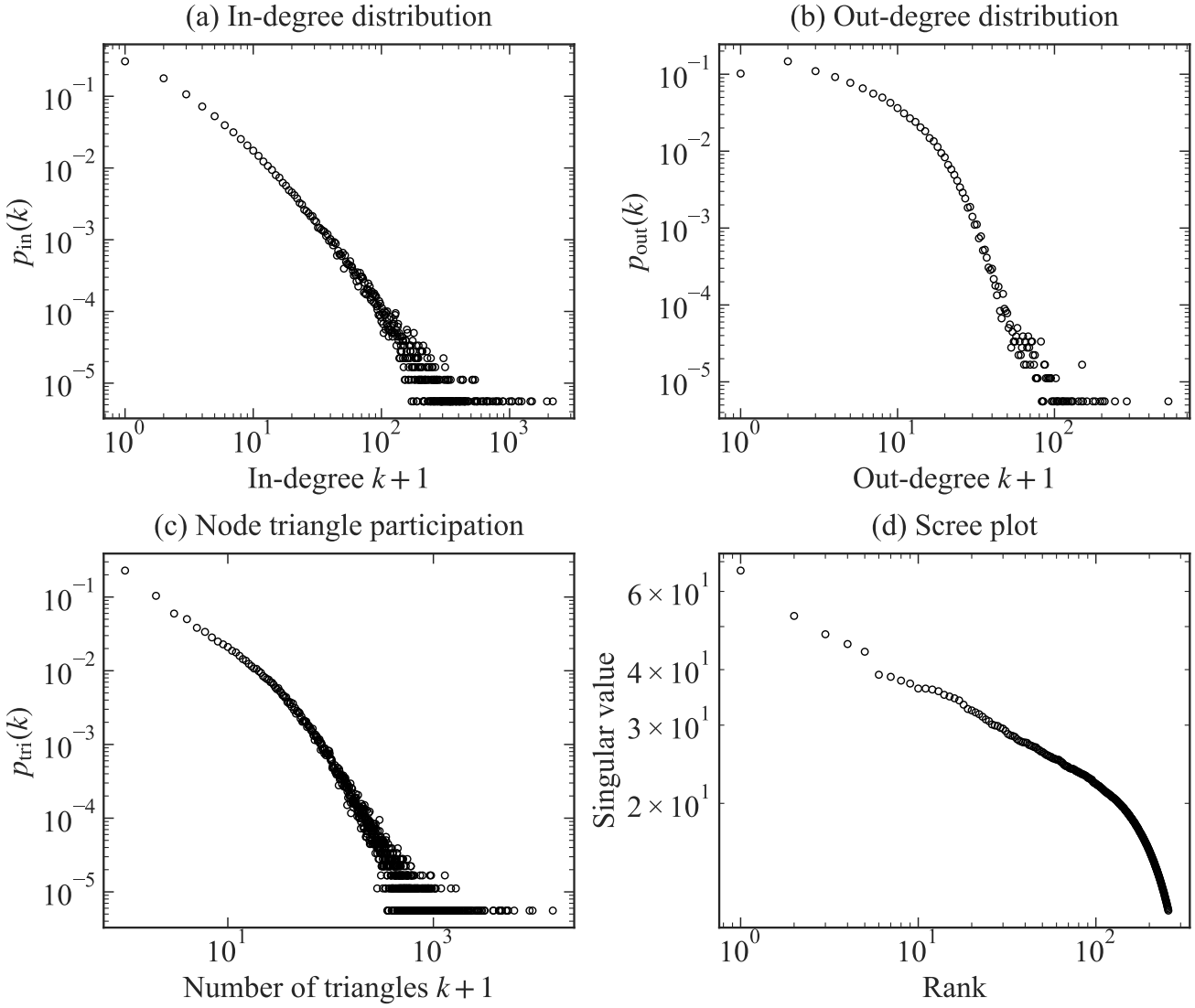


図 3.5. `WoS-Stat` のネットワーク特徴量．(a) 入次数分布 p_{in}, (b) 出次数分布 p_{out}, (c) 三角形数の分布 p_{tri}, (d) scree plot

3.3　時刻に依存したネットワーク特徴量

引用ネットワークには，古い文献は出次数が小さく，新しい文献は出次数が小さいといった特徴をもつ．これは古い文献が引用するより古い文献はデータの範囲外となること，新しい文献を引用するより新しい文献はデータの範囲外となることが原因となる．モデリングにはこれらの性質は望ましくなく，全てのノードで同じ性質となるような性質を用いるべきである．我々は時刻に依存しないネットワーク特徴量を定義してモデリングに用いた．

3.3.1　時間調整した引用の年齢分布

まずある文献 $v_i \in V$ から引用した文献 $v_j \in A_{\mathrm{out}}(v_i)$ の時刻差 $s = \tau(v_i) - \tau(v_j)$ を引用年齢 (citation age) s と定義する．さらに時刻 t における時刻年齢 s ごとの引用数

$$m(s,t) = |\{u \mid v \in V,\ \tau(v) = t,\ u \in A_{\mathrm{out}}(v),\ \tau(v) - \tau(u) = s\}|$$

は用いて，時刻 t における時刻年齢 s ごとの引用年齢分布 (citing age distribution) $c(s,t)$ は $c(s,t) = m(s,t)/n(t)$ と定義できる．ここで $n(t) = |\{v \mid v \in V,\ \tau(v) = t\}|$ である．引用年齢分布については文献 [Redner, 2004, Golosovsky and Solomon, 2017] で議論されており，さらに文献 [Hajra and Sen, 2005] では 2 種類の年齢分布，引用年齢分布と非引用年齢分布について取り扱っている．

3.3.2　時間調整した出次数の分布

同様に引用年齢 s を考慮して出次数を

$$d_{\mathrm{out}}(v,s) = |\{u \mid u \in A_{\mathrm{out}}(v),\ \tau(v) - \tau(u) = s\}|\,.$$

と改める．ここで $0 \leq s \leq \tau(v) - 1$ であることを用いると $d_{\mathrm{out}}(v)$ は

$$d_{\mathrm{out}}(v) = \sum_{s=0}^{\tau(v)-1} d_{\mathrm{out}}(v,s)$$

と $d_{\mathrm{out}}(v,s)$ の和であることが分かる．ここでいくつかの例外 ($s < 0$ もしくあ $\tau(v) \leq s$) を無視していることに注意されたい．このように $d_{\mathrm{out}}(v)$ は $\tau(v)$ に強く依存していることが明らかである．例えば $\tau(v) = 1$ となる $d_{\mathrm{out}}(v)$ はほぼ 0 となる．そのため $c(s,t)$ が t にほぼ依存しないことを前提に $d_{\mathrm{out}}(v)$ を次のように修正した．$d^T_{\mathrm{out}}(v)$ は

$$d^T_{\mathrm{out}}(v) = \sum_{s=0}^{T-1} \left(d_{\mathrm{out}}(v,s) \frac{\sum_{i=0}^{T-1} c(i)}{\sum_{i=0}^{s} c(i)} \right)$$

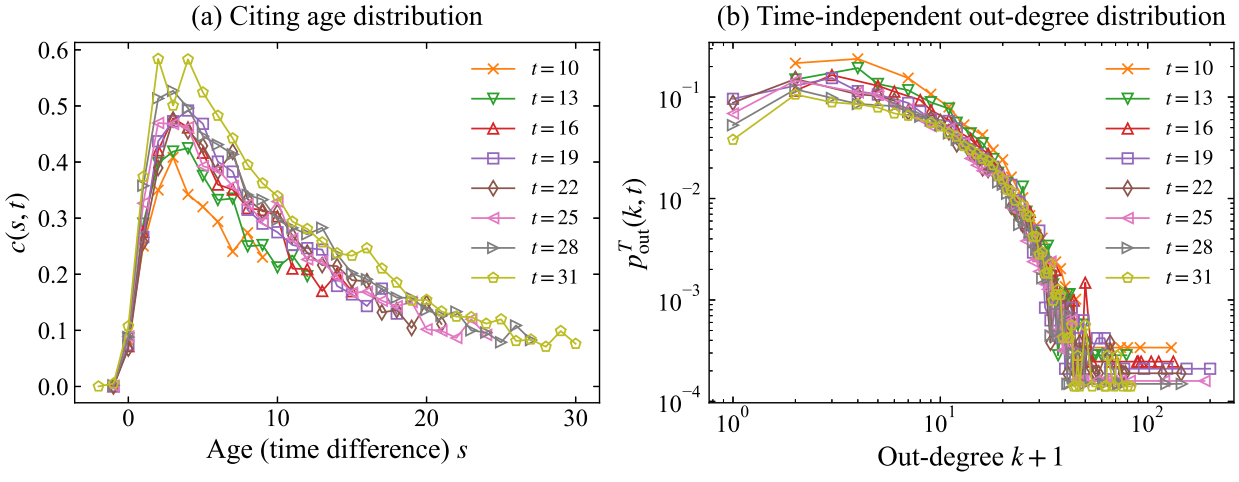


図 3.6. `WoS-Stat` における時間調整された特徴量: (a) 時刻 t における時刻差 s からなる引用年齢分布 $c(s,t)$, (b) 時刻 t における次数 k からなる時刻調整された出次数分布 $p^T_{\text{out}}(k,t)$

と定義する．ここで $c(s) = \frac{1}{T-s}\sum_{t=s+1}^{T} c(s,t)$ である．$d^T_{\text{out}}(v)$ を時間調整された出次数 (time-adjusted out-degrees) と呼ぶことにする．そして同様に次数 k に対応する時間調整された出次数分布 (time-adjusted out-degree distribution) を $p^T_{\text{out}}(k) = \frac{\left|\{v|v\in V,\ d^T_{\text{out}}(v)=k\}\right|}{|V|}$ と定義する．

図 3.6 は引用元の発表時刻 t と引用年齢 s ごとの引用年齢分布 $c(s,t)$ と，引用元の発表時刻 t ごとの時間調整された出次数分布 $p^T_{\text{out}}(k,t) = \frac{\left|\{v|v\in V,\ \tau(v)=t,\ d^T_{\text{out}}(v)=k\}\right|}{n(t)}$ を可視化したものである．引用元の発表時間は $t \in \{10,13,16,19,22,25,28,31\}$ のみをプロットしている．いずれの特徴量 $c(s,t)$, $p^T_{\text{out}}(v,t)$ も引用元の発表時刻 t に独立した性質をもつことを確認できる．

3.4 提案モデル

3.4.1 いくつか特徴量のモデル化

引用ネットワークに対する提案モデルはいくつかの要素を含んでいるが，まずはモデルが用いる特徴量について説明を行う．本モデルはこれらの分布を入力とするため,

まず各時刻 t における文献数 $n(t)$ の期待値が次に示すロジスティクス関数 $f_n(t)$ で近似できると仮定する．

$$f_n\left(t \mid \mu_n, \sigma_n, \kappa_n\right) = \frac{\kappa_n}{1+\exp(-\frac{t-\mu_n}{\sigma_n})}$$

ここで，生成される文献数は by $\lfloor f_n(t) + \epsilon_n(t) \rfloor$ となる．ここで $\epsilon_n(t)$ は $N(0,\eta_n^2)$ に従う独立な確率変数であるとする．そして $\lfloor x \rfloor$ は実数 x を超えない最大の整数値とする．

なお文献 [Hajra and Sen, 2005] では同様の目的のために $f_n\left(t \mid a,b\right) = a(1-\exp(-bt))$ を用いているものの, 少なくとも `WoS-Stat` に対しては十分ではないと考えている．

続いて年齢分布 $c(s)$ の期待値は定数 κ_c 倍された逆ガウス分布 (inverse Gaussian distribu-

tion) [Seshadri, 1999] の確率密度関数 (PDF) で近似できると仮定する．

$$f_c\left(s \mid \gamma_c, \mu_c, \sigma_c, \kappa_c\right) = \frac{\kappa_c}{\sigma_c\sqrt{2\pi\left(\frac{s-\mu_c}{\sigma_c}\right)^3}} \exp\left(-\frac{\left(\frac{s-\mu_c}{\sigma_c} - \gamma_c\right)^2}{2\gamma_c^2\left(\frac{s-\mu_c}{\sigma_c}\right)}\right).$$

なお文献 [Wu and Holme, 2009] においては同様の目的に指数分布で近似できるとしている．しかしながら `WoS-Stat` において，年齢分布 $c(s)$ の形状は公開直後では相対的に低く，その後急激に増加してピークをむかえ，その後，徐々に減少するという特徴を有しており，指数分布で表現が難しい．

そして調整済みの出次数 d_{out}^T は一般化パレート分布 (generalized Pareto distribution) [Hosking and Wallis, 1987] に従う確率変数であると仮定する．一般化パレート分布の確率密度関数 (PDF) は

$$f_o\left(x \mid \gamma_o, \mu_o, \sigma_o\right) = \frac{1}{\sigma_o}\left(1 + \gamma_o\frac{x-\mu_o}{\sigma_o}\right)^{-1-\frac{1}{\gamma_o}}$$

となる．なお一般化パレート分布 は $\gamma_o = 0$ と $\mu_o = 0$ となる場合，指数分布 (exponential distribution) と一致する．指数分布の確率密度関数は

$$f_o\left(x \mid \mu_o, \sigma_o\right) = \frac{1}{\sigma_o}\exp\left(-\frac{x-\mu_o}{\sigma_o}\right)$$

となる．一般化パレート分布は表現力が高いものの，`WoS-Stat` に対しては指数分布でも十分な適合を得られる．

3.4.2 生成プロセスのモデル化

提案モデルの最後要素は（引用）エッジ生成の機構である．関数 f_n, f_c, f_o をもとづき優先的選択 (Preferential attachment; PA) 機構と三角形形成 (Triad formation; TF) 機構を組み合わせる．時刻 t でサイズが $\lfloor f_n(t) + \epsilon_n(t) \rfloor$ のノードが生成される．その各ノードは f_o にもとづいて生成された x に従い次数 $k = \lfloor x \rceil$ をもち，後述する PA 機構もしくは TF 機構によりエッジ k が生成される．なお PA 機構と TF 機構はそれぞれ確率 $1-\beta$ と確率 β で確率的に選択される．

ノード v_i がネットワークに追加され，出次数 k が割り当てられるとする．PA 機構では v_i はすでにネットワークに存在するノード集合からノード $v_j \in V$ を確率

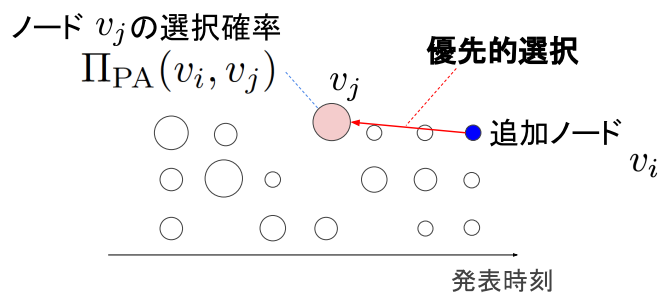
$$\Pi_{\text{PA}}(v_i, v_j) \propto Im(v_j) \cdot f_c(\tau(v_i) - \tau(v_j)) \tag{3.1}$$

で選択する．このとき $Im(v_j)$ は v_j の重要度を表し，$f_c(\tau(v_i) - \tau(v_j))$ は時刻差 $\tau(v_i) - \tau(v_j)$ における文献引用率の経年変化を示している．一方で 機構では v_i は直前の PA 機構で選択された v_j の周辺ノード集合 $A(v_j)$ から $v_k \in A(v_j)$ を確率

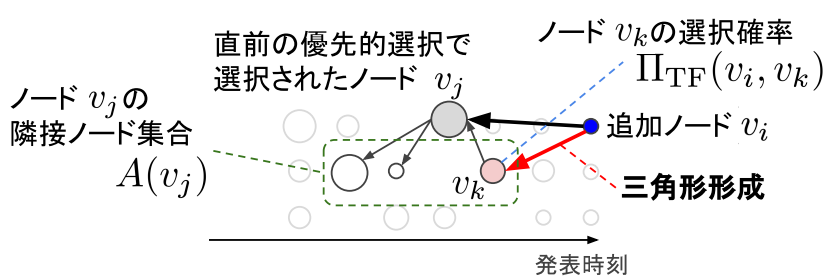
$$\Pi_{\text{TF}}(v_i, v_k) \propto Im(v_k) \cdot f_c(\tau(v_i) - \tau(v_k)), \tag{3.2}$$

で選択する．その後，v_i の出次数 k で指定された回数，PA 機構もしくは TA 機構を繰り返す．論文の重要度を決定することは難しいため，ノード v の重要度 $Im(v)$ を $d_{\text{in}}(v) + 1$ で近似することとす

る．図 3.7a と 3.7b はそれぞれ我々の生成モデルで用いる PA 機構と TF 機構を図示したものである．



(a) 優先的選択 (Preferential attachment; PA) 機構



(b) 三角形形成 (Triad formation; TF) 機構

図 3.7. 提案モデルの PA 機構と TF 機構

3.5 関数群の推定とシミュレーション

一般にグラフ生成モデルの適合性を検証することは困難であるため，本研究でシミュレーションで得られた実現値による検証を実施する．前述のように我々のモデルがもついくつかの構造に基づいて，可能な限り正確にシミュレーションを実施した．

3.5.1 パラメータの推定

`WoS-Stat` に対して, $f_n(t)$, $f_c(s)$, f_o を推定する方法を説明する．まず f_n は時刻 $t \in \{1, 2, ..., T\}$ ごとの文献数 $n(t)$ を用いて最小自乗法により推定し, 推定されたパラメータ $\hat{\mu}_n = 33.263$, $\hat{\sigma}_n = 14.743$, $\hat{\kappa}_n = 17242.068$, $\hat{\eta}_n = 328.047$ を得た．

続いて f_c は `WoS-Stat` から得られた時刻差 $s \in \{0, 1, ..., T-1\}$ に対する調整済み時刻差分布 $c(s)$ を用いて最小自乗法により推定する．推定の安定性を向上するため，時刻の範囲 $t \geq 10$ のみに限定し用いる．そのため $c(s)$ を時刻 t ごとに分解した $c(s, t)$ を用いる．これは図 3.6 においても形状の安定性が確認できる．パラメータ $\hat{\gamma}_c = 2.509$, $\hat{\mu}_c = -1.427$, $\hat{\sigma}_c = 14.361$, $\hat{\kappa}_c = 10.191$ を得た．

f_o が想定する指数分布に従う確率変数は本来，連続値を取ることになるが，我々は各 v の時刻調整された出次数分布 $d^T_{\mathrm{out}}(v)$ をデータとして用いて，最尤推定によりパラメータを推定した．f_c の推定と同様に $d^T_{\mathrm{out}}(v)$ を $d^T_{\mathrm{out}}(v, t)$ に分解し，$t \geq 10$ の範囲で用いた．パラメータ $\hat{\mu}_o = 0.000$,

$\hat{\sigma}_o = 8.116$ を得た．

図 3.8 は WoS-Stat において推定された関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ とそれに対応するデータ $n(t)$, $c(s)$, p^T_{out} を比較したものである．いずれの関数も十分に推定ができていると判断することができる．

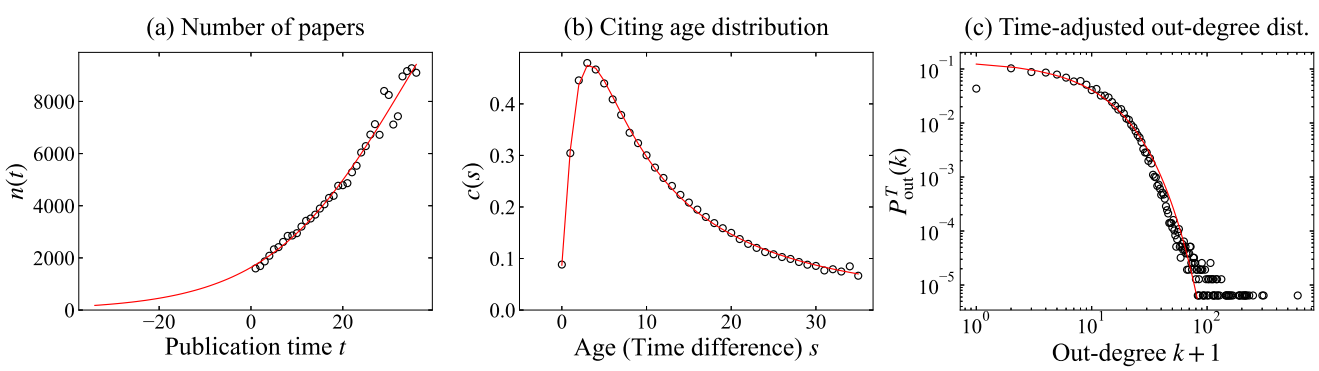


図 3.8. WoS-Stat における推定された関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ (赤線) とデータ (黒丸) の比較．

3.5.2 ネットワーク生成のためのシミュレーション

提案モデルのシミュレーションも，他の成長モデル（Barabási–Albert, Holme–Kim, Wu–Holme）における時間ごとにノードやエッジを追加していく類似の手順を用いる．

まずノード集合 V' とエッジ集合 E' を空集合 $\emptyset$ で初期化する．その後, 時刻 t を $-T+1$, $-T+2$, ..., T と 1 ずつ変化させて，後続の処理を実行する．まず時点 t において，アルゴリズムはサイズが $\lfloor f_n(t) + \epsilon_n(t) \rfloor$ となるノード集合を V' に追加する．そのとき $t \geq 0$ であれば追加された各ノード v_i に対して PA 機構や TF 機構によるエッジ生成を行い，$t \leq 0$ であれば何もしない．追加するエッジ数 k は f_o に基づいて生成した乱数 x の整数部分 $k = \lfloor x \rfloor$ を用いる．

エッジ生成の 1 回目は必ず PA 機構が実行される．2 回目以降は実行時に指定されたパラメータ β を用いて，PA 機構を確率 $1-\beta$ で，TF 機構を確率 β で決定する．

PA 機構は式 (3.1) に基づいてノード v_j を選択し，エッジ (v_i, v_j) を E' に追加する．まず時刻差 $s \in \{0, 1, \ldots, T-1\}$ を $f_c(s)$ に比例する確率で決定し，その後，ノード $v_j \in \{v \mid v \in V',\ \tau(v_i) - \tau(v) = s\}$ を $d_{\mathrm{in}}(v_j) + 1$ に比例する確率で選択する．

一方, TF 機構では，式 (3.2) に基づいてノード v_j の周辺ノード集合 $W(v_i, v_j, s)$ からノード v_k を選択し，ノード v_i とのエッジ (v_i, v_k) を E' に追加する．まず直前の PA 機構で選択した v_j と隣接する周辺ノード集合を，v_i との時刻差 s ごとに以下のように構成する．

$$W(v_i, v_j, s) = \{v \mid v \in A(v_j),\ \tau(v_i) - \tau(v) = s\} \setminus \{v_i\} \tag{3.3}$$

その後，時刻差 s を $f_c(s)$ に比例する確率で選択し，$v_k \in W(v_i, v_j, s)$ を $d_{\mathrm{in}}(v_k) + 1$ に比例する確率で選択する．ここで $W(v_i, v_j, s)$ が空集合になる s に対しては $f_c(s) = 0$ とする．全ての $W(v_i, v_j, s)$ が空集合である場合，TF 機構の代わりに PA 機構を実行する．

最後に, V' や E' のうち，時刻の範囲外となるノードやエッジを除外した V, E を出力する．

$$V = \{v \mid v \in V',\ 1 \leq \tau(v) \leq T\} \tag{3.4}$$

$$E = \{(v_i, v_j) \mid v_i, v_j \in V,\ (v_i, v_j) \in E'\} \tag{3.5}$$

以上の手順を付録の Algorithm 1 としてまとめた．このシミュレーション手順では時刻の範囲外である $-T+1 \leq t \leq 0$ に対してノードを生成し，各時刻において T 期間遡ったエッジ生成を行う．既存モデルの初期状態を小さな連結成分とする場合に比べて，この手順はより自然な引用構造を実現できると考えている．

Algorithm 1: GenerateCitationNetwork

Input: 各時刻 $t \in \{-T+1, ..., T\}$ ごとの文献数 $f_n(t)$, 各時刻差 $s \in \{0, ..., T-1\}$ ごとの時刻分布 $f_c(s)$, 時刻調整された出次数分布 f_o, TF 機構の実行割合 β

Result: (V, E)

Procedure `PreferentialAttachment`(V, f_c, v_i)**:**
- 時刻差 $s \in \{0, 1, ..., T-1\}$ を $f_c(s)$ に比例する確率で選択
- ノード $v_j \in \{v \in V \mid s = \tau(v_i) - \tau(v)\}$ を $d_{\text{in}}(v_j) + 1$ に比例する確率で選択
- **return** v_j

Procedure `TriadFormation`(f_c, v_i, v_j)**:**
- 時刻差 $s \in \{0, 1, ..., T-1\}$ ごとに $W(v_i, v_j, s) = \{v \mid v \in A(v_j),\ \tau(v_i) - \tau(v) = s\} \setminus \{v_i\}$ を構成
- 時刻差 $s \in \{0, 1, ..., T-1\}$ を $f_c(s)$ に比例する確率で選択．ただし $W(s)$ が $\emptyset$ である場合は確率は 0 とする．
- ノード $v_k \in W(v_i, v_j, s)$ を $d_{\text{in}}(v_k) + 1$ に比例する確率で選択
- **return** v_k

Procedure `GenerateCitationNetwork`(T, f_n, f_c, f_o, β)**:**
- (V', E') を $(\emptyset, \emptyset)$ で初期化
- **for** $t_i \leftarrow -T+1$ **to** T **do**
  - $U \leftarrow \{v_{|V'|+i} \mid i \in \{1, 2, .., \lfloor f_n(t_i) \rfloor\}$
  - $V' \leftarrow V' \cup U$
  - **if** $t_i < 0$ **then**
    - **contitnue**
  - **foreach** $v_i \in U$ **do**
    - $v_j \leftarrow \emptyset$
    - f_o に従う乱数 r を生成し, $k \leftarrow \lfloor r \rceil$ とする
    - **for** 1 **to** k **do**
      - **if** $v_j \neq \emptyset$ **or** `Random(0,1)` $< \beta$ **then**
        - $v_k \leftarrow$ `TriadFormation`(f_c, v_i, v_j)
        - **if** $v_k \neq \emptyset$ **then**
          - $E' \leftarrow E' \cup \{(v_i, v_k)\}$
          - **contitnue**
      - $v_j \leftarrow$ `PreferentialAttachment`(V', f_c, v_i)
      - $E' \leftarrow E' \cup \{(v_i, v_j)\}$
- $V \leftarrow \{v \mid v \in V',\ 1 \leq \tau(v) \leq T\}$
- $E \leftarrow \{(v_i, v_j) \mid v_i, v_j \in V,\ (v_i, v_j) \in E'\}$
- **return** (V, E)

3.6 WoS-Stat に対するシミュレーションによるモデル適合の確認

3.6.1 各関数の推定結果の検討

図 3.9 は WoS-Stat に対するシミュレーション結果の $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ についてプロットである．各図は 10 回のシミュレーションにおける平均値と近似的な 95% 信頼区間を表している．いずれもデータとシミュレーション結果が近くあてはまりを確認することができた．

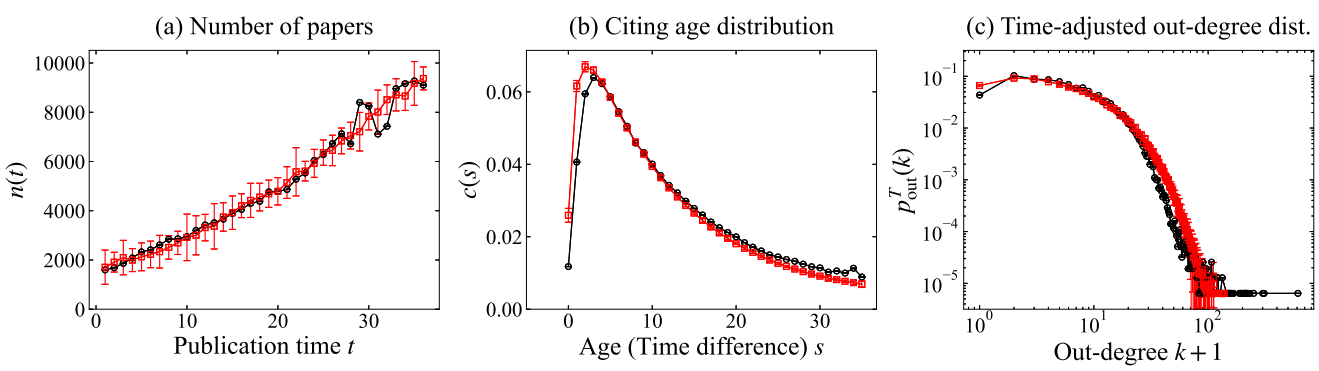


図 3.9. WoS-Stat に対するシミュレーションの結果 (red squares and error bars) とデータ (black circles) の比較．(a) 時刻ごとの文献数 $n(t)$ と $\hat{f}_n$, (b) 引用年齢分布 $c(s)$ と $\hat{f}_c$, (c) 時刻調整された出次数分布 p^T_{out} と $\hat{f}_o$

3.6.2 パラメータ β の調整

モデルにおいて, TF 機構を実行する確率を制御するパラメータである β は推定することが難しい．そのため我々はシミュレーションにより最適な値を決定する．シミュレーションでは β を 0.85 から 0.99 まで 0.01 で変化させ，データとの最も類似するところを決定する．なおこのパラメータの決定方法は Wu–Holme と同じ方法である [Wu and Holme, 2009]．入次数分布，出次数分布，三角形数の分布に対して，それぞれデータ WoS-Stat との Kullback–Leibler (K–L) divergence を類似度の指標とする．より正確には対数でヒストグラム化したデータ列に対する K-L 情報量を用いた．いずれの分布も裾が長い分布をとるため，対数での類似度を評価を行うためである．図 3.10 は β ごとに 10 回のシミュレーションで得られた K-L 情報量の平均値と近似的な 95% 信頼区間をプロットしたものである．結果から β が変化しても入次数や出次数には影響がなく，三角形数が変化していることがわかる．特に $0.93 \leq \beta \leq 0.95$ 辺りで減少し，最小値は $\beta = 0.94$ となった．なお $\beta = 0.94$ と決定しても，各ノードでのエッジ生成のうち 1 回目は必ず PA が実行されるため，TF 機構の実行割合は β よりも小さくなることは注意されたい．

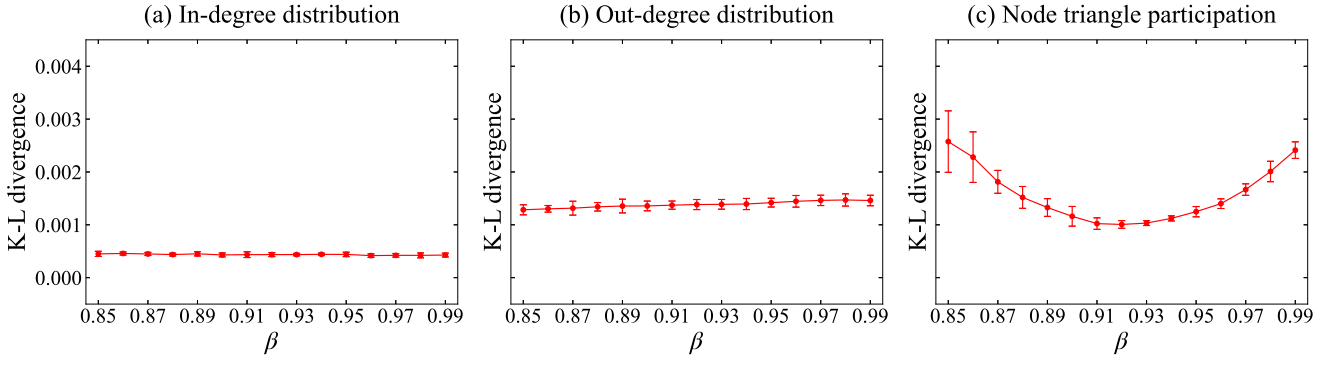


図 3.10. β を変化させたときの，データ `WoS-Stat` に対するシミュレーション結果の Kullback–Leibler 情報量: β を変化させても (a) 入次数や (b) 出次数は大きく変化しない．一方で (c) 三角形数の分布はアーチ状の形状となり, 最小値は $\beta = 0.94$ となる．

3.6.3 ネットワーク特徴量を用いたモデル適合の確認

引用ネットワークの解明に適したネットワーク特徴量の可視化により，モデルの適合を検証する．ネットワーク特徴として，入次数分布と外次数分布，三角形数の分布，および scree plot を用いる．これらは文献 [Leskovec et al., 2010] でモデル適合の検証に使用されている．

なお比較対象の既存モデルとして Barabási–Albert モデル [Barabási and Albert, 1999], Holme–Kim モデル [Holme and Kim, 2002], Wu–Holme モデル [Wu and Holme, 2009] を用いる．なお Barabási–Albert モデルと Holme–Kim モデルは出次数を定数で指定する必要があり，`WoS-Stat` の出次数の平均値である 6 を用いた．加えて Holme–Kim モデルや Wu–Holme モデル，提案モデルは TF 機構を実行する確率に $\beta = 0.94$ を指定した．これは提案モデルと同じ設定である．ここで Wu–Holme モデルは時刻順にならんだノード ID とそれに紐付く出次数，ノード ID 差に基づく年齢分布（これは）が必要になる．文献集合を文献 ID と発表年の昇順で並び替え，文献 ID を $1, 2, ..., |V|$ と振り直して，出次数列やノード ID 差の頻度から推定した年齢分布を入力とした．

提案モデルのシミュレーション・アルゴリズムは Python の NetworkX ライブラリ [Hagberg et al., 2008] を用いて実装している．参照実装は付録 B.4 にまとめる．なおパラメータの推定には SciPy [Virtanen et al., 2020] を用いた．比較対象の Barabási–Albert モデルや Holme–Kim モデルについては NetworkX の `barabasi_albert_graph` と `powerlaw_cluster_graph` を用いた．Wu–Holme モデルは `powerlaw_cluster_graph` を参考に実装した．推定は SciPy を用いた．ネットワーク特徴量については SNAP package [Leskovec and Sosič, 2016] を用いて算出した．

図 3.11 は引用ネットワーク `WoS-Stat` に対して各モデルが生成したネットワーク構造をまとめた．まず全てのモデルは入次数分布について良いあてはまりを示している．これはいずれのモデルも PA 機構を採用しているためと考えられる．しかしながら，より厳密には Barabási–Albert モデルと Holme–Kim モデルでは入次数にもとづいた PA 機構を用いているのに対し，Wu–Holme モデ

ルではノード ID 差に基づく引用の年齢分布による PA 機構を採用している．また提案モデルでは次数と，離散時刻にもとづく引用の年齢分布をどちらも考慮した PA 機構を採用しているものの，入次数分布ではそれらの違いはあまり確認できず，いずれも適合している．出次数分布については Barabási–Albert モデルや Holme–Kim モデルにいくつか課題が確認できるが，これらのモデルが出次数を定数と仮定しているためである．一方で Wu–Holme モデルは非常に高い適合を示したものの，これは出次数列に対してデータそのものを入力しているためである．提案モデルは，出次数列そのものは必要とせず，時間調整した出次数分布を用いた．三角形数の分布については，三角形型の引用構造を考慮していない Barabási–Albert モデルで課題が確認できる．Holme–Kim モデルについてはある程度大きな k については一定のあてはまりを示しているものの，小さい k については差が大きい．それらに対して Wu–Holme モデルは高いあてはまりを確認することができた．Holme–Kim モデルと Wu–Holme モデルの差については，TF 機構に差がないため，PA 機構における時刻差の考慮の影響と考えられる．提案モデルでは Wu–Holme モデルと同等の適合を示していることが確認できた．Scree plot については提案モデルは他のモデルよりも若干離れている．その一方で比較対象のモデルのうち，いずれもしある程度狭い領域にプロットされており，大きな問題はないと考えている．

このように提案モデルは (a) と (b) と (c) で十分に適合するものの，(d) については若干の課題が存在するという結果となった．一方で Wu–Holme モデルでは (a), (b), (c), (d) についていずれも高い適合を示しているものの，論文の発表時刻でソートされたノード ID が必要になること，またシミュレーションに出次数列そのものを必要となる点に課題があると考えている．提案モデルはそのどちらも必要がない完全なシミュレーションを実現できている．また提案モデルで扱う時刻情報は Wu–Holme モデルに比べてかなり粗い粒度となることにも注意されたい．

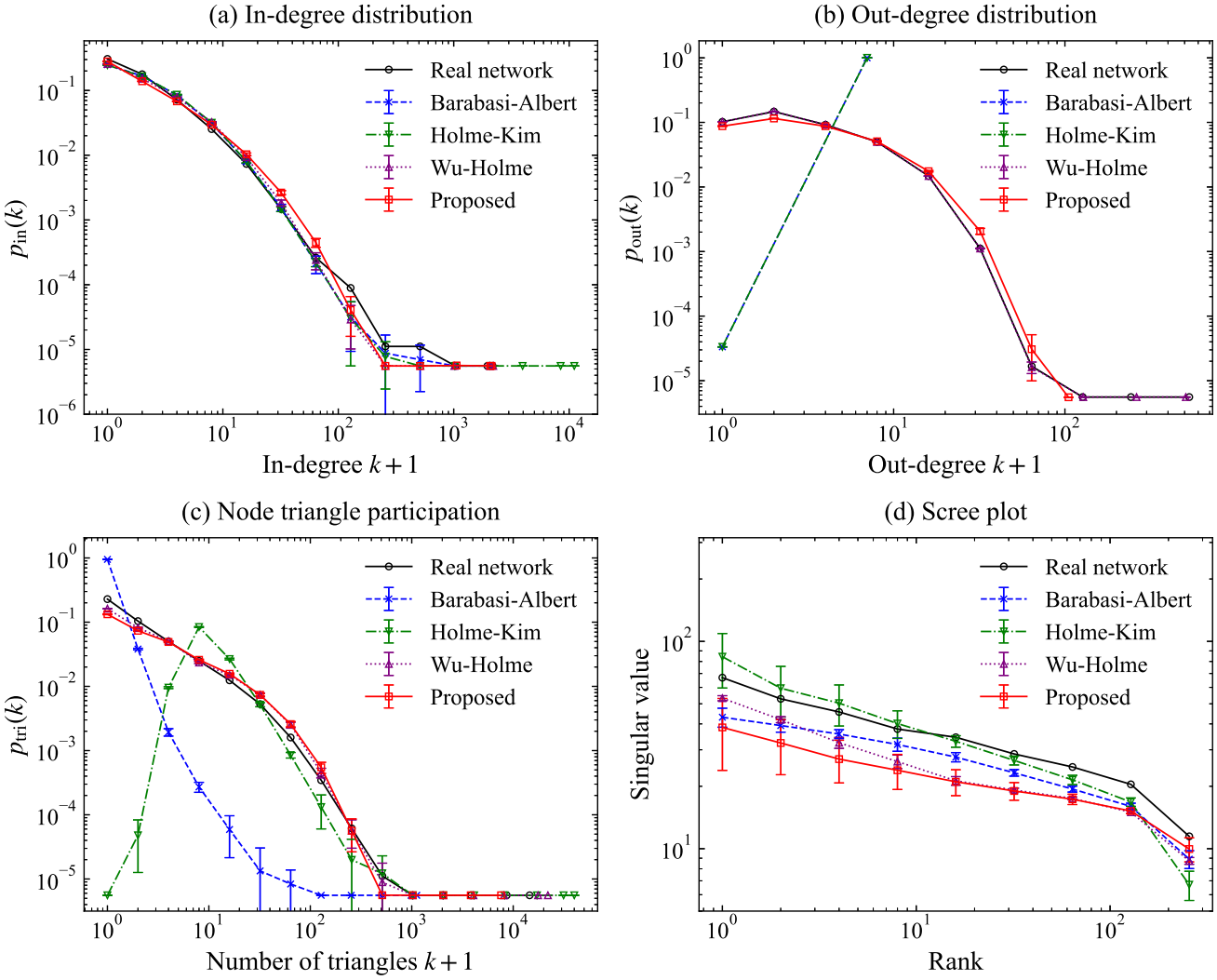


図 3.11. 引用ネットワーク `WoS-Stat` とシミュレーション結果の比較. (a) 入次数分布, (b) 出次数分布, (c) 三角形数の分布, (d) Scree plot.

3.7　arXiv 引用ネットワークを用いた検証

提案モデルの有用性を示すため，arXiv [Cornell University, 1991] の書誌データから生成された引用ネットワークに対して，WoS-Stat と同様の検証を行う．用いた arXiv-HepTh と arXiv-HepPh はそれぞれ arXiv [Cornell University, 1991] の書誌データの高エネルギー物理分野 hep-th と hep-ph から生成された引用ネットワークである．これらは SNAP プロジェクト [Leskovec and Krevl, 2014] で公開されている．arXiv-HepTh は文献 [Hajra and Sen, 2005, Wu and Holme, 2009, Leskovec et al., 2010] で分析や生成モデルの検証の対象にしている．arXiv-HepPh は文献 [Leskovec et al., 2010] で分析されている．

特に arXiv-HepTh はいくつかの文献 [Hajra and Sen, 2005, Wu and Holme, 2009, Leskovec et al., 2010] で arXiv-HepPh は文献 [Leskovec et al., 2010] ですでにモデリングの対象となるが，いずれも年次の分析である．

3.7.1　引用ネットワークの構築

hep-th は各レコードに「引用元文献 ID，引用先文献 ID」となる引用データのほか，「文献 ID，発表時刻」となる公開時刻データが公開されている．一方 hep-ph は引用データのみで，公開時刻データを持たない．そこでいずれに対しても各レコードに「引用元文献 ID，引用先文献 ID」となる引用データのみを用いて，文献 ID から抽出可能な時刻情報を用いて，引用ネットワーク arXiv-HepTh と arXiv-HepPh を構築する．

どちらのデータも文献 ID は YYMMNNN という形式となり，このうち YY は発表年の下二桁に，MM は発表月と発表時刻情報に対応する．しかしながら一部，さらに先頭のゼロの欠損が推測されたため YYMMNNN に従ってゼロを補完する．補完に関する詳細は付録 B.2 にまとめる．

発表時刻に年次を用いた場合は arXiv-HepPh が 11 期間，arXiv-HepTh が 10 期間となり，期間数が十分でない．そこで arXiv-HepPh が 44 期間，arXiv-HepTh が 40 期間となる四半期間を用いる．表 3.3 は arXiv-HepTh と arXiv-HepPh の規模や期間をまとめた．

表 3.3. arXiv の高エネルギー物理分野の引用ネットワーク arXiv-HepTh and arXiv-HepPh.

Instance	文献数 （ノード数）	引用数 （エッジ数）	期間
arXiv-HepTh	27,770	352,285	1992/01–2002/12 (11 年分, 44 四半期間)
arXiv-HepPh	34,546	421,578	1993/01–2002/12 (10 年分, 40 四半期間)

3.7.2　パラメータの推定

図 3.12 と図 3.13 は arXiv-HepTh と arXiv-HepPh に対し，$\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ のパラメータ推定を行い，得られたパラメータの関数をまとめたものである．

arXiv-HepTh での推定は $\hat{f}_n$ が $\hat{\mu}_n = 6.556$, $\hat{\sigma}_n = 10.779$, $\hat{\kappa}_n = 802.889$, and $\hat{\eta}_n = 56.448$ に，$\hat{f}_c$ は $\hat{\gamma}_c = 5103.305$, $\hat{\mu}_c = -1.402$, $\hat{\sigma}_c = 8.132$, and $\hat{\kappa}_c = 1.704$ に，$\hat{f}_o$ が $\hat{\gamma}_o = 0.031$, $\hat{\mu}_o = 0.000$, and $\hat{\sigma}_o = 15.209$ となった．データの時刻の区間の後半では $\hat{f}_n$ が少し乖離があるものの，開始から中盤までのあてはまりは良い．

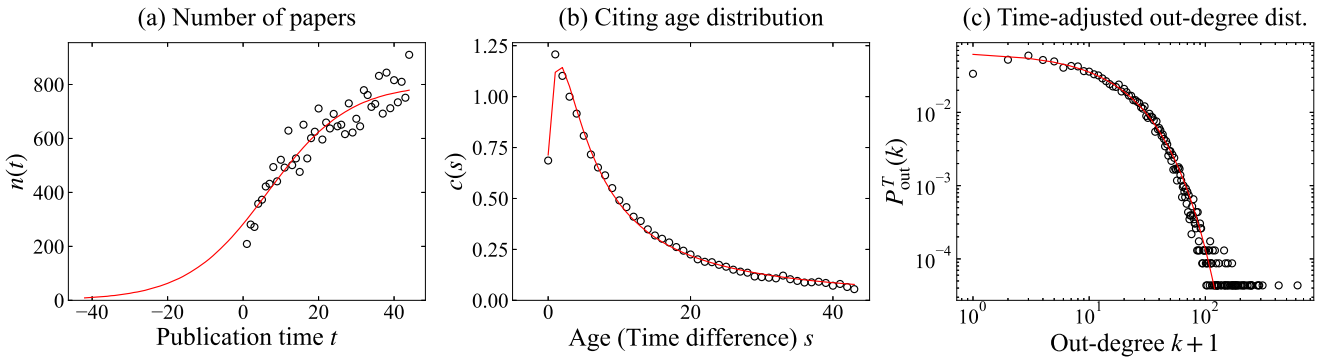


図 3.12. arXiv-HepTh における推定された関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ (赤線) とデータ (黒丸) の比較.

arXiv-HepPh での推定は $\hat{f}_n$ が $\hat{\mu}_n = 5.775$, $\hat{\sigma}_n = 7.416$, $\hat{\kappa}_n = 1050.755$, and $\hat{\eta}_n = 85.154$ に，$\hat{f}_c$ が $\hat{\gamma}_c = 9195.910$, $\hat{\mu}_c = -2.179$, $\hat{\sigma}_c = 13.150$, and $\hat{\kappa}_c = 2.040$ に，f_o が $\hat{\gamma}_o = 0.032$, $\hat{\mu}_o = 0.000$, and $\hat{\sigma}_o = 14.496$ となった．こちらはいずれのあてはまりも問題がない．$\hat{f}_c$ の形状を比較してみると arXiv-HepTh に比べて arXiv-HepPh は，ピークが高く，減衰が早い性質がある．

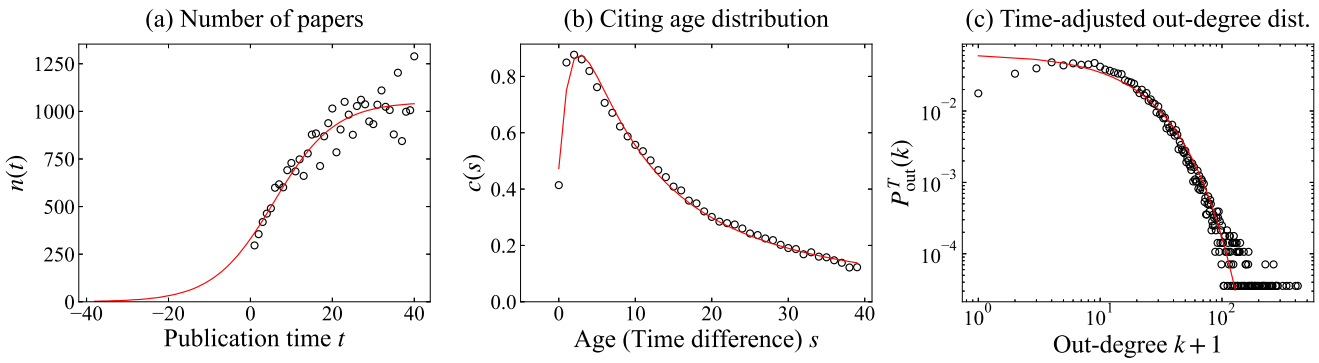


図 3.13. arXiv-HepPh における推定された関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ (赤線) とデータ (黒丸) の比較.

3.7.3　ネットワーク特徴量を用いたあてはまりの確認

図 3.14 と図 3.15 は arXiv-HepTh と arXiv-HepPh のネットワーク特徴量をまとめたものである．この引用ネットワークに対しても既存モデルとの比較を行う．WoS-Stat での結果と同様に，提案モ

デルは入次数分布，出次数分布，三角形数の分布，いずれの分布に対しても良い当てはまりを示した．さらに scree plot については `WoS-Stat` の場合と比べて，既存モデルよりもあてはまり結果が良好であることを示している．

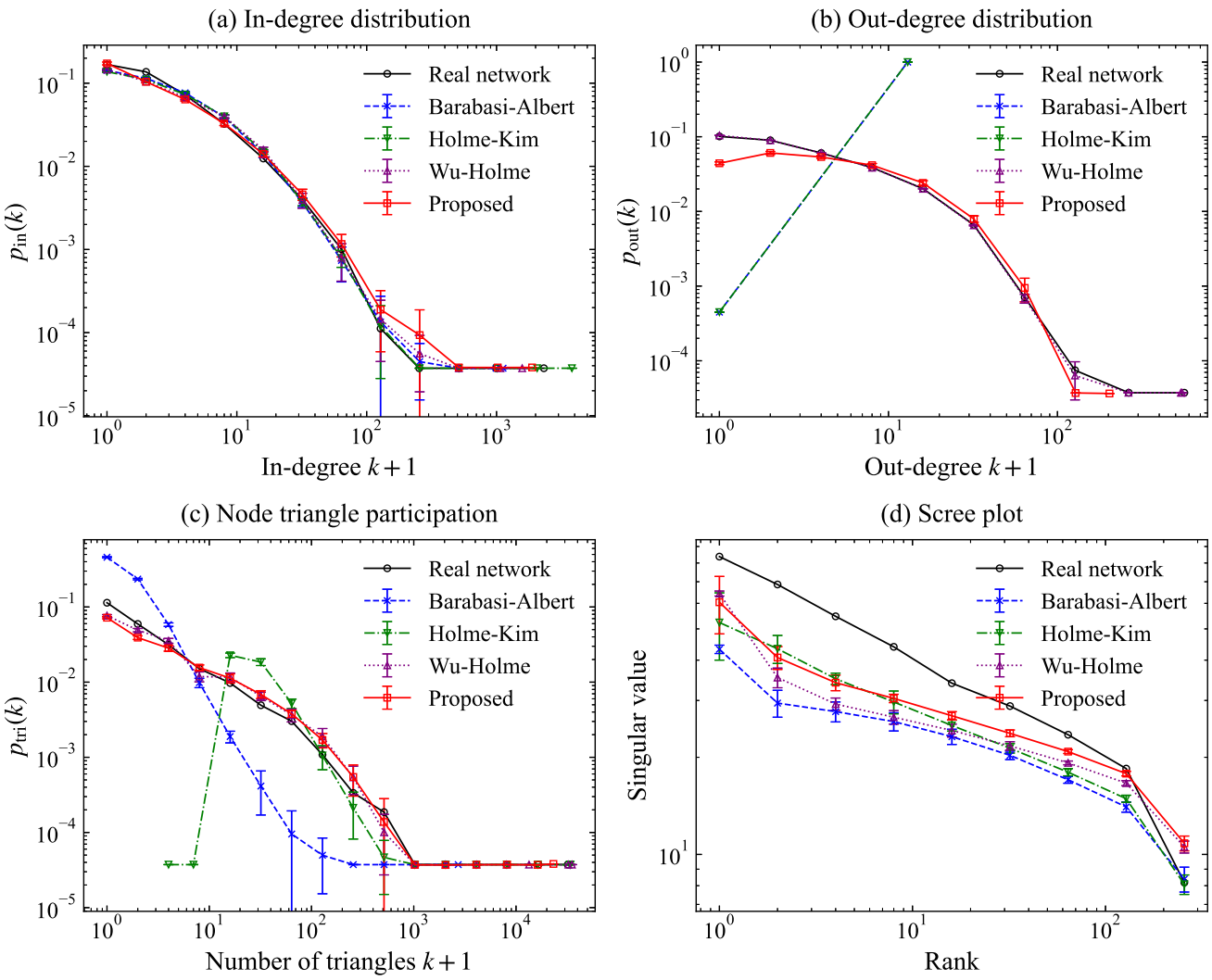


図 3.14. 引用ネットワーク `arXiv-HepTh` とシミュレーション結果の比較．(a) 入次数分布, (b) 出次数分布, (c) 三角形数の分布, (d) Scree plot.

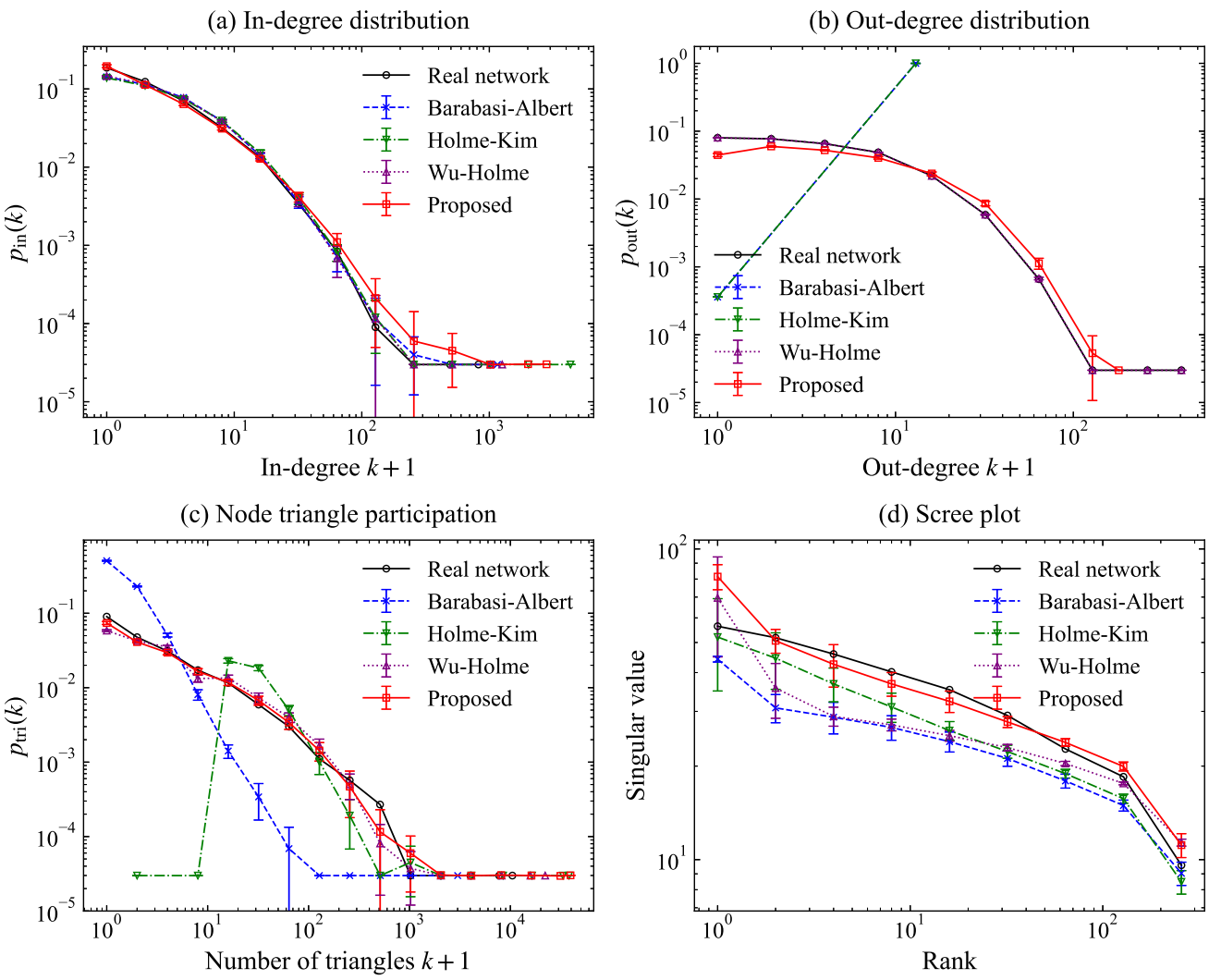


図 3.15. 引用ネットワーク `arXiv-HepPh` とシミュレーション結果の比較. (a) 入次数分布, (b) 出次数分布, (c) 三角形数の分布, (d) Scree plot.

3.8　シミュレーション結果の解釈

本章では，引用ネットワークを表現するグラフの確率的生成モデルを提案した．対象は Web of Science 書誌データの統計確率分野から生成された引用ネットワーク WoS-Stat である．時刻ごとの論文数の推移，引用年齢分布，時刻調整済み出次数分布はデータにおいて変化しないと仮定した．この仮定はある程度データによって裏付けられており，パラメータを正確に推定するために必要である．しかしながら論文の引用構造は急速に変化しているため，将来的には変化する可能性も十分に考えられる．

モデルを定義する関数として，ロジスティック関数，逆ガウス確率密度関数，指数分布を採用した．いずれもデータを近似するため用いたが，これらの関数の意味を解釈したり理論的に検証することは困難である．その一方で，これらの関数が元のデータに近いデータを生成するために有益であることを確認することである．

提案モデルでは PA 機構と TF 機構を用いており，特に PA 機構では論文の重要度を近似するために入次数を採用している．しかしながら論文の真の重要度は潜在的な変数であり，複雑なモデルで推定する必要がある．また出次数を論文の種類を近似的に表すと捉えている．ここで十分に古い論文への引用はデータに含まれないため，論文の出次数は小さくなる傾向にある．一方で論文の出次数が小さいほどその論文が他の分野に注力しているとも解釈できる．このように十分に古い論文への引用を多くもつ論文と，他分野に焦点を当てた論文が同じタイプだと捉えてしまう点が課題といえる．

入次数と出次数は 2 ノード間の関係を考慮し，三角形は 3 ノード間の関係を考慮した特徴量である．提案モデルは最大 3 ノード間の関係を明示的に考慮しており，この仮定のもとで提案モデルがグラフの生成プロセスを単純かつ十分な近似となることを示した．その一方でシミュレーションされたグラフの scree plot はデータのものとは比較的離れてしまったが，これは scree plot が 3 ノード以上の関係を表現しているためと解釈できる．

このモデルの重要な特徴は離散時間を明示的に考慮し，離散時間情報をグラフ構造上で解釈しやすい点である．またデータの期間外となる過去の時点を考慮したエッジ生成を行うことで，実際の状況に近いシミュレーションを実行することができる．範囲外のノードやエッジはアルゴリズムの最終段階で，実データと同様に破棄される．これは他の提案モデル [Barabási and Albert, 1999], [Holme and Kim, 2002], [Wu and Holme, 2009] との違いである．いずれのモデルも初期状態を小さな連結成分で近似し，その連結性を維持したまま成長させる．そのため，生成されるグラフ構造は提案モデルとは異なり，常に連結成分となる．提案モデルは，他の引用ネットワーク arXiv-HepTh と arXiv-HepPh に対しても有効であることを実証することができた．このように提案モデルは一般的な引用ネットワークに対し，良い近似を提供することが期待できる．

第 4 章

特許文献の引用ネットワークに対する確率生成モデル

3 章で取り扱った生成モデルは学術論文の引用構造を対象とした．本章では特許文献の引用ネットワークに対して該当モデルを適用すると，あてはまりが十分でないことが明らかとなった．適合を改善するために該当モデルを拡張し，より一般的な生成モデルを提案する．

4.1　特許文献の引用ネットワーク cit-Patents

本研究で用いる特許文献の引用ネットワーク cit-Patents について説明を行う．このデータは全米経済研究所 (National Bureau of Economic Research; NBER) で公開される書誌データをもとに構築された引用ネットワークである．書誌データには 1975 年から 1999 年の米国特許データから抽出された，3,774,768 件の文献とそれらの間に生成された 16,518,948 件の引用が含まれる．さらにメタ情報として特許文献ごとに文献 ID，事前定義されたカテゴリとサブカテゴリ，文献 ID のリストで構成される参考文献が利用可能である．ここでメタ情報をもつ文献間の引用だけに限定し引用ネットワークを構築すると，2,075,770 ノードと 10,557,536 エッジのネットワーク構造を得られる．引用ネットワークの構築に関する詳細は付録 B.3 にまとめる．

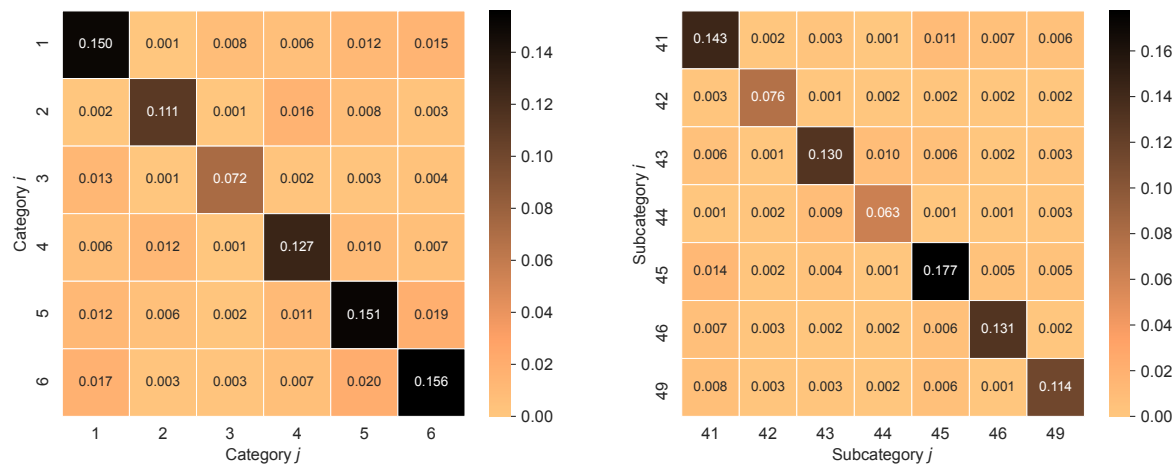
4.1.1　カテゴリとサブカテゴリによる階層的なクラスタ構造

特許文献の引用ネットワーク cit-Patents の各ノードに付与されたカテゴリやサブカテゴリについて性質を説明する．まず表 4.1 は cit-Patents におけるカテゴリとサブカテゴリの関係，サブカテゴリごとの文献数をまとめたものである．カテゴリは 4-9 のサブカテゴリを含み，それぞれのサブカテゴリは数万から数十万程度の文献を含む．つづいて図. 4.1a はネットワーク全体を対象にカテゴリ内外のエッジ数をヒートマップで可視化したものである．各 $(i, j)-$ 成分はカテゴリ i からカテゴリ j までの相対的な引用数を表している．対角成分（76.764%）はそれ以外の成分に比べて明らかに割合が大きく，各カテゴリがクラスタを形成していることが確認できる．さらに図. 4.1b はカテゴリ

4 に着目し，その内部のサブカテゴリ内外の引用数の割合をまとめたものである．カテゴリの場合と同様に，対角成分（76.895%）はそれ以外の成分に比べて明らかに割合が大きく，各サブカテゴリもカテゴリ同様にクラスタの形成を確認することができる．

表 4.1. 全米経済研究所で公開される書誌データのカテゴリとサブカテゴリ

カテゴリ	サブカテゴリ	カテゴリ名	サブカテゴリ名	文献数
1	11	Chemical	Agriculture,Food,Textiles	25 624
1	12	Chemical	Coating	44 366
1	13	Chemical	Gas	14 331
1	14	Chemical	Organic Compounds	124 981
1	15	Chemical	Resins	100 725
1	19	Chemical	Miscellaneous-chemical	296 907
2	21	Computers & Communications	Communications	122 981
2	22	Computers & Communications	Computer Hardware & Software	91 614
2	23	Computers & Communications	Computer Peripherials	24 282
2	24	Computers & Communications	Information Storage	51 460
3	31	Drugs & Medical	Drugs	84 824
3	32	Drugs & Medical	Surgery & Med Inst.	70 573
3	33	Drugs & Medical	Biotechnology	32 170
3	39	Drugs & Medical	Miscellaneous-Drgs&Med	16 632
4	41	Electrical & Electronic	Electrical Devices	99 950
4	42	Electrical & Electronic	Electrical Lighting	46 950
4	43	Electrical & Electronic	Measuring & Testing	84 098
4	44	Electrical & Electronic	Nuclear & X-rays	42 880
4	45	Electrical & Electronic	Power Systems	103 534
4	46	Electrical & Electronic	Semiconductor Devices	52 603
4	49	Electrical & Electronic	Miscellaneous-Elec	69 726
5	51	Mechanical	Mat. Proc & Handling	167 725
5	52	Mechanical	Metal Working	94 679
5	53	Mechanical	Motors & Engines + Parts	109 459
5	54	Mechanical	Optics	64 848
5	55	Mechanical	Transportation	88 856
5	59	Mechanical	Miscellaneous-Mechanical	155 811
6	61	Others	Agriculture,Husbandry,Food	63 994
6	62	Others	Amusement Devices	29 619
6	63	Others	Apparel & Textile	55 158
6	64	Others	Earth Working & Wells	43 822
6	65	Others	Furniture,House Fixtures	61 256
6	66	Others	Heating	40 733
6	67	Others	Pipes & Joints	27 151
6	68	Others	Receptacles	63 173
6	69	Others	Miscellaneous-Others	256 427



(a) 引用ネットワーク内のカテゴリ構造　　**(b)** カテゴリ 4 内のサブカテゴリ構造

図 4.1. NBER の特許引用ネットワークにおけるカテゴリ内・カテゴリ間の引用数の割合と，カテゴリ 4 内のサブカテゴリ内・サブカテゴリ間の引用数の割合

4.1.2　グラフ・クラスタリングを用いた階層構造の確認

データに含まれるカテゴリやサブカテゴリによる階層的な密構造を，ネットワーク上の性質と対応させて解釈が可能であるかを確認する．対象の引用ネットワーク全体に対してクラスタリングを実施して，どのようなコミュニティ構造（密構造）が得られるか検証を行った．用いたクラスタリング・アルゴリズムは Louvain 法 [Blondel et al., 2008] である．Louvain 法はコミュニティ構造を評価するための指標であるモジュラリティ [Girvan and Newman, 2002] を最大化するという方針でコミュニティ構造（密構造）を検出する．実装は NetworKit [Staudt et al., 2016] の `networkit.community.detectCommunities` を用いた．Louvain 法は無向グラフを前提としているため，引用ネットワークを無向グラフに変換して適用した．

モジュラリティ最大化によるクラスタリング手法は各クラスタが同じ性質（ノードの次数の期待値が同等である）という仮定が存在するが，今回は次数分布には大きな差がないため問題なくクラスタ構造が抽出されることが期待される．一方，より高いクラスタリング性能を求めるためには確率的ブロックモデル [Karrer and Newman, 2011] などが存在するものの，対象のネットワーク規模の観点から適用は難しい．

図 4.2 はクラスタリングで得られたコミュニティを，そのサイズ（該当するノード数）順にプロットしたものである．得られたコミュニティの個数は合計 181,521 個であるが，その多くは 10 ノード以下の小さいコミュニティである．また上位 36 クラスタで全体の 90.475 % と一部の大きなクラスタとそれ以外に分かれている．

図 4.3 と図 4.4 に，得られたコミュニティのうちサイズが上位 36 件に対して，カテゴリやサブカテゴリとの関係をまとめた．縦軸をカテゴリやサブカテゴリ，横軸を得られたコミュニテとして，各要素には分類された文献集合の大きさを表している．さらにコミュニティを考慮した局所性を確認する

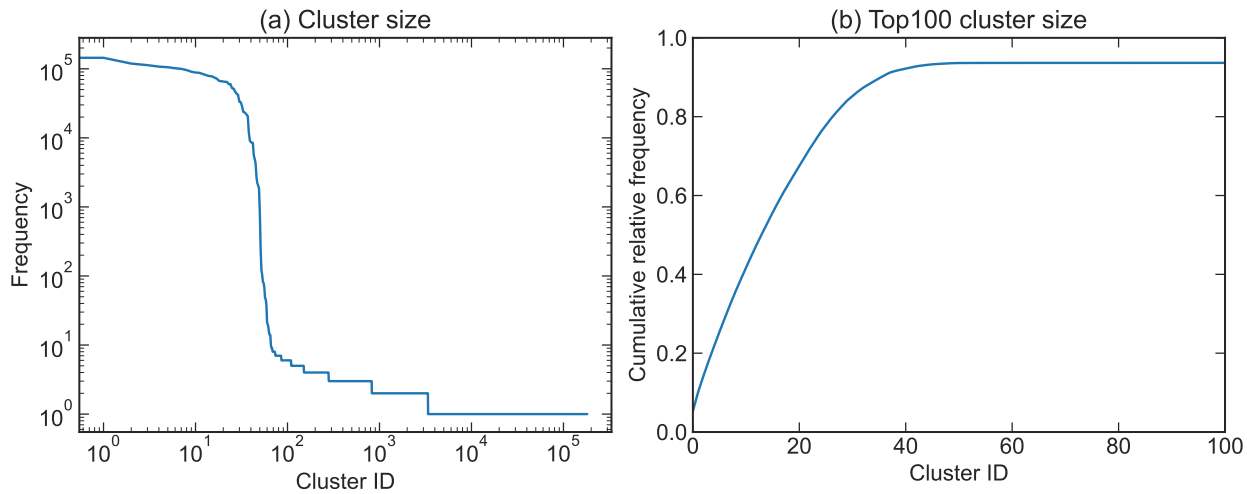


図 4.2. グラフ・クラスタリングで得られたコミュニティの数とサイズ

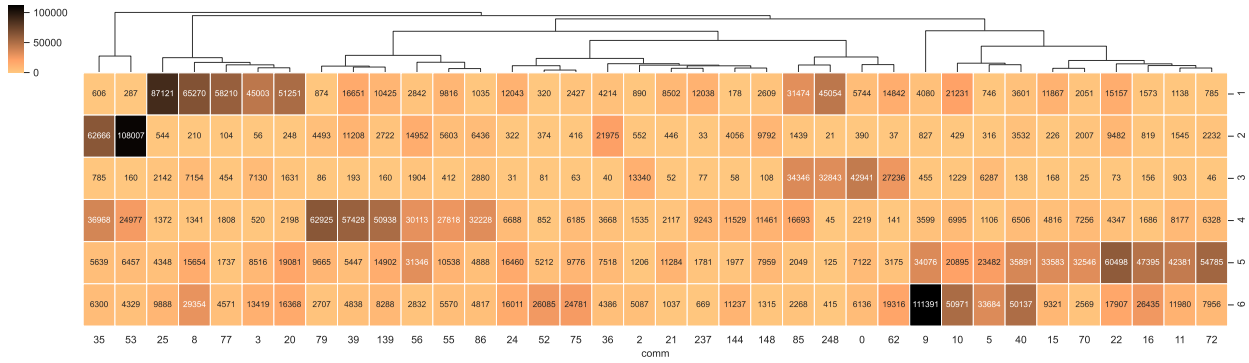


図 4.3. グラフ・クラスタリングで得られた上位 36 のコミュニティとカテゴリの関係

ため，横軸（コミュニティ）に対し階層的クラスタリングを実施した．図より，まず得られたコミュニティ構造は，カテゴリよりも細かく，いくつかのサブカテゴリと同等程度の粒度であることが推測できる．特にコミュニティ 35, 53 はそれぞれサブカテゴリ 21, 22 に紐付いており，グラフクラスタリングによる密構造とサブカテゴリの粒度が一致している．その一方で，サブカテゴリ 14, 15, 19 は，いくつかのコミュニティ 3, 8, 20, 25, 62, 77, 85, 248 に混在している．このようにおおよそサブカテゴリ程度の粒度であり，一部のカテゴリについてはいくつかのクラスタで説明ができる．

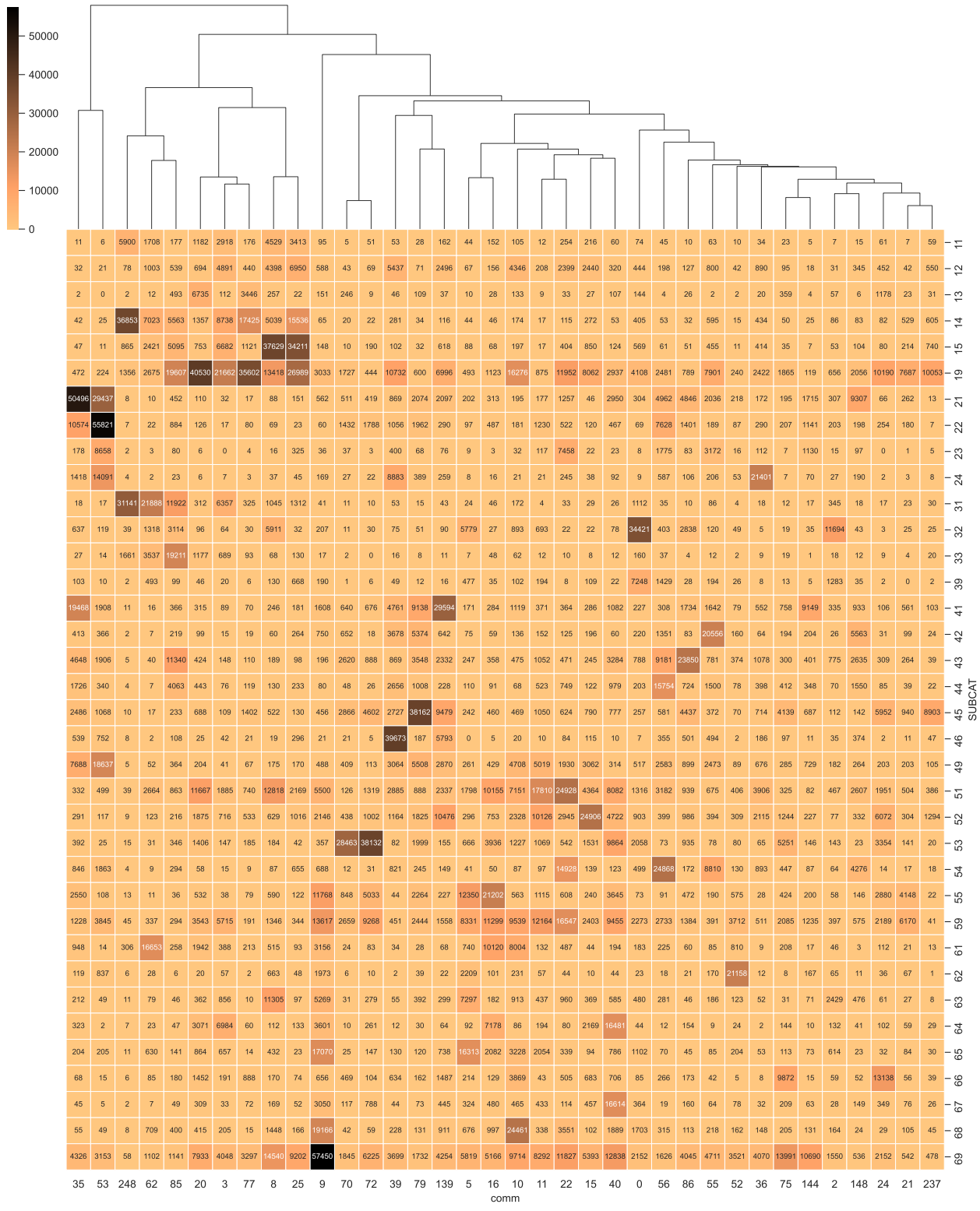


図 4.4. グラフ・クラスタリングで得られた上位 36 のコミュニティとサブカテゴリの関係

4.1.3　カテゴリ構造とサブカテゴリ構造のネットワーク特徴量

3 章で取り扱った生成モデルは単一分野内のネットワークを想定するため，階層的なクラスタ構造をもつネットワーク全体へのあてはまりは期待できない．まずはカテゴリごと，サブカテゴリごとにサブグラフを形成し，それらのネットワーク特徴量を比較する．

図 4.5 はネットワーク全体とカテゴリごとのネットワーク特徴量をまとめたものである．(a) 入次数分布, (b) 出次数分布, (c) 三角形数の分布については，ネットワーク全体と，各カテゴリごとのサブグラフのネットワーク特徴量は近い性質をもっている．さらに図 4.6, 4.7, 4.8, 4.9, 4.10, 4.11 は，それぞれカテゴリ 1, 2, .., 6 のカテゴリ全体とサブカテゴリのネットワーク特徴量となる．全体とカテゴリの関係と同様に，各カテゴリの全体と，カテゴリに含まれる各サブカテゴリについても，ネットワーク特徴量に類似性が高いことが確認できる．

以上の観察から，各カテゴリやサブカテゴリは全体と比べて，ネットワーク特徴量に関して大きな性質の差はないことが明らかになった．そこで以降の分析ではサブカテゴリ 41 に関連するサブグラフ `cit-Patents-sc41` に着目する．

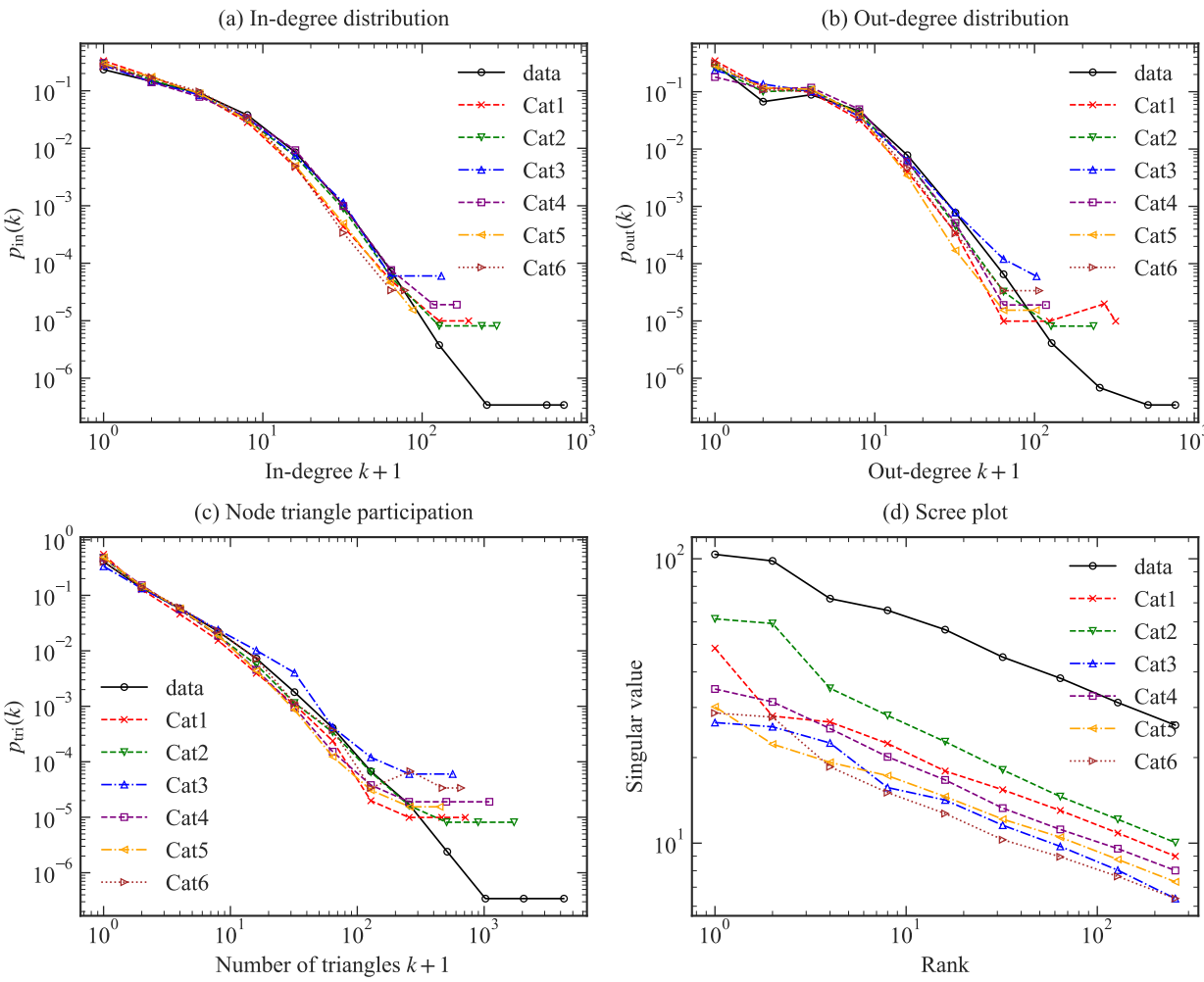


図 4.5. 全体とカテゴリごとのネットワーク特徴量

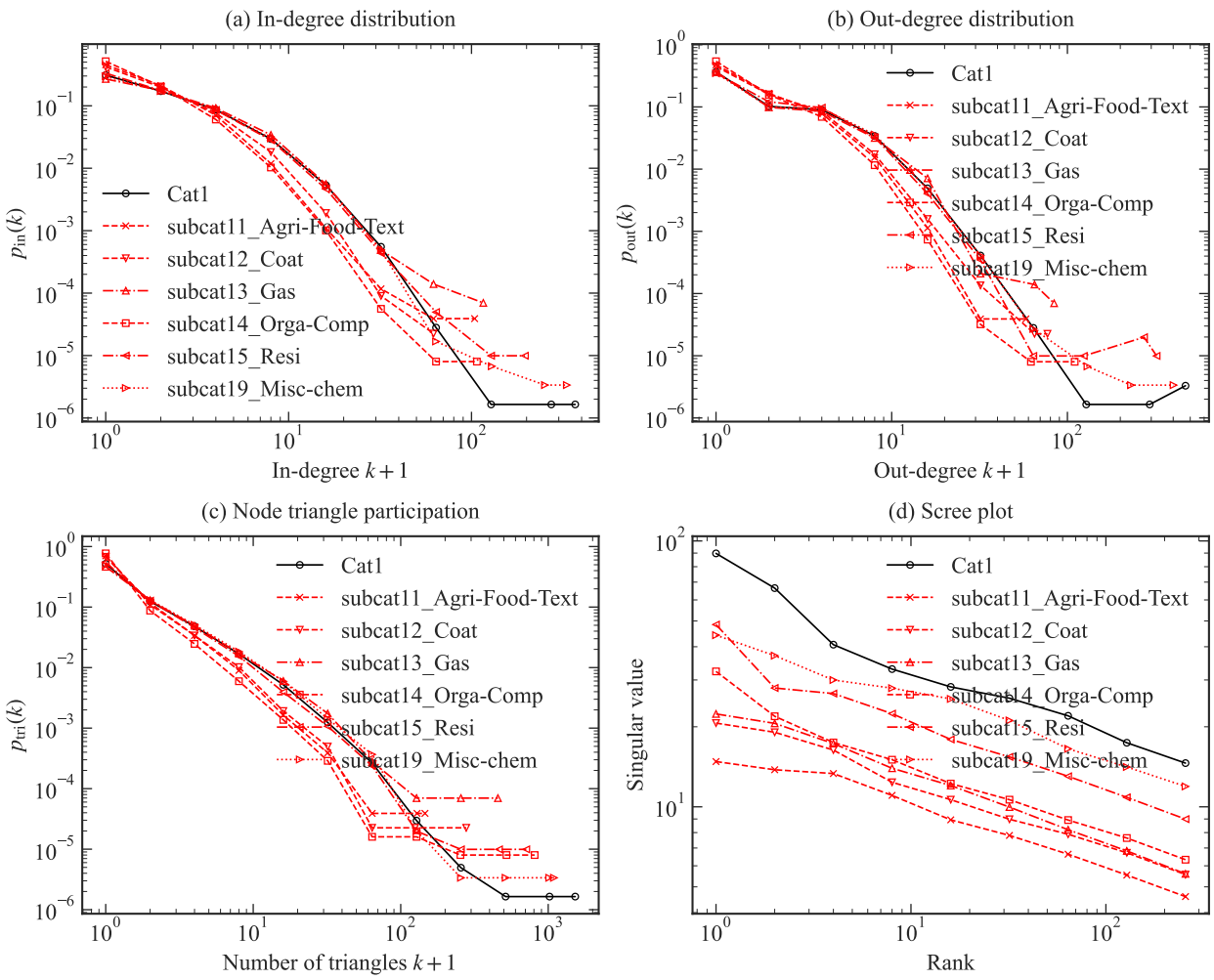


図 4.6. カテゴリ 1 のネットワーク特徴量

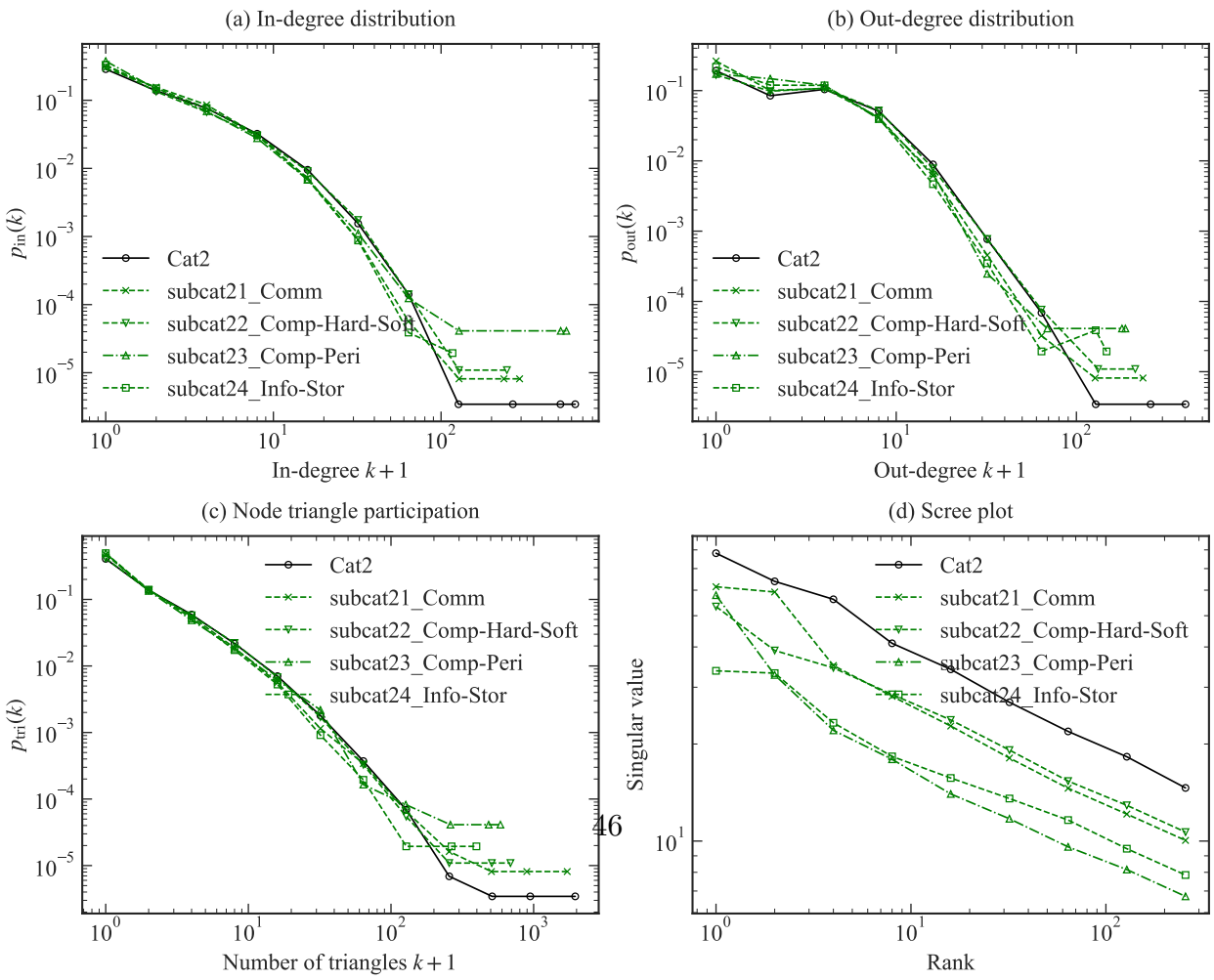


図 4.7. カテゴリ 2 のネットワーク特徴量

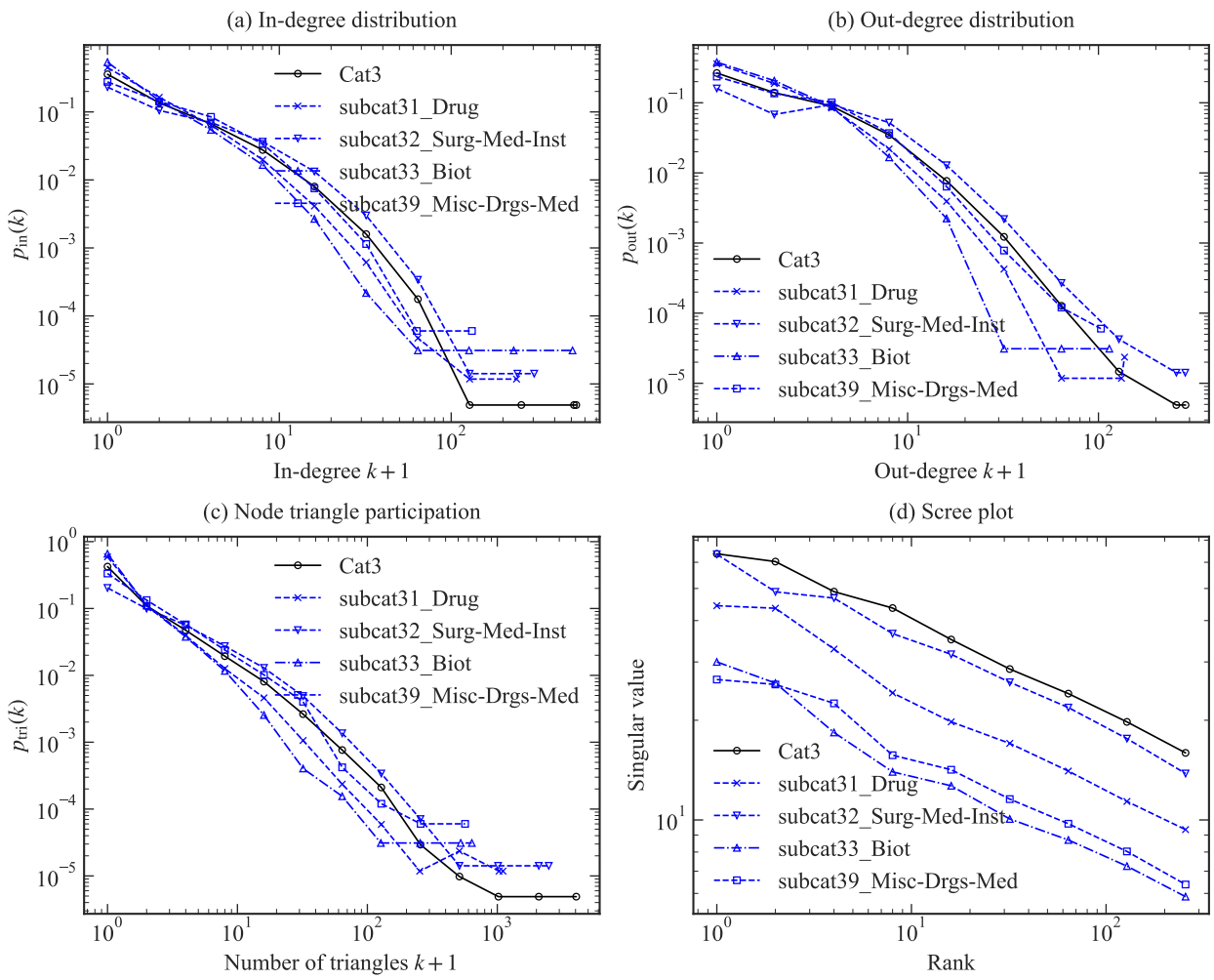


図 4.8. カテゴリ 3 のネットワーク特徴量

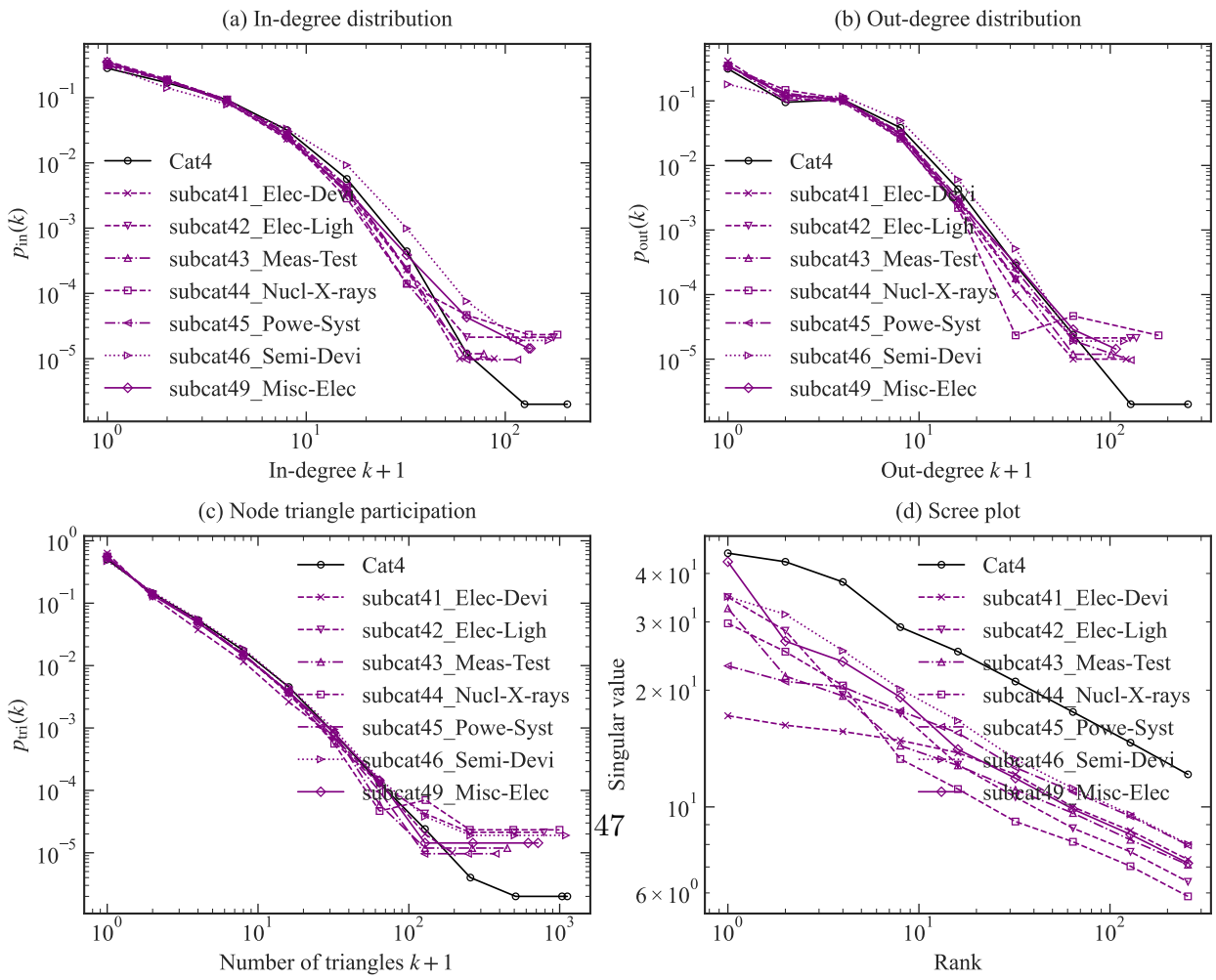


図 4.9. カテゴリ 4 のネットワーク特徴量

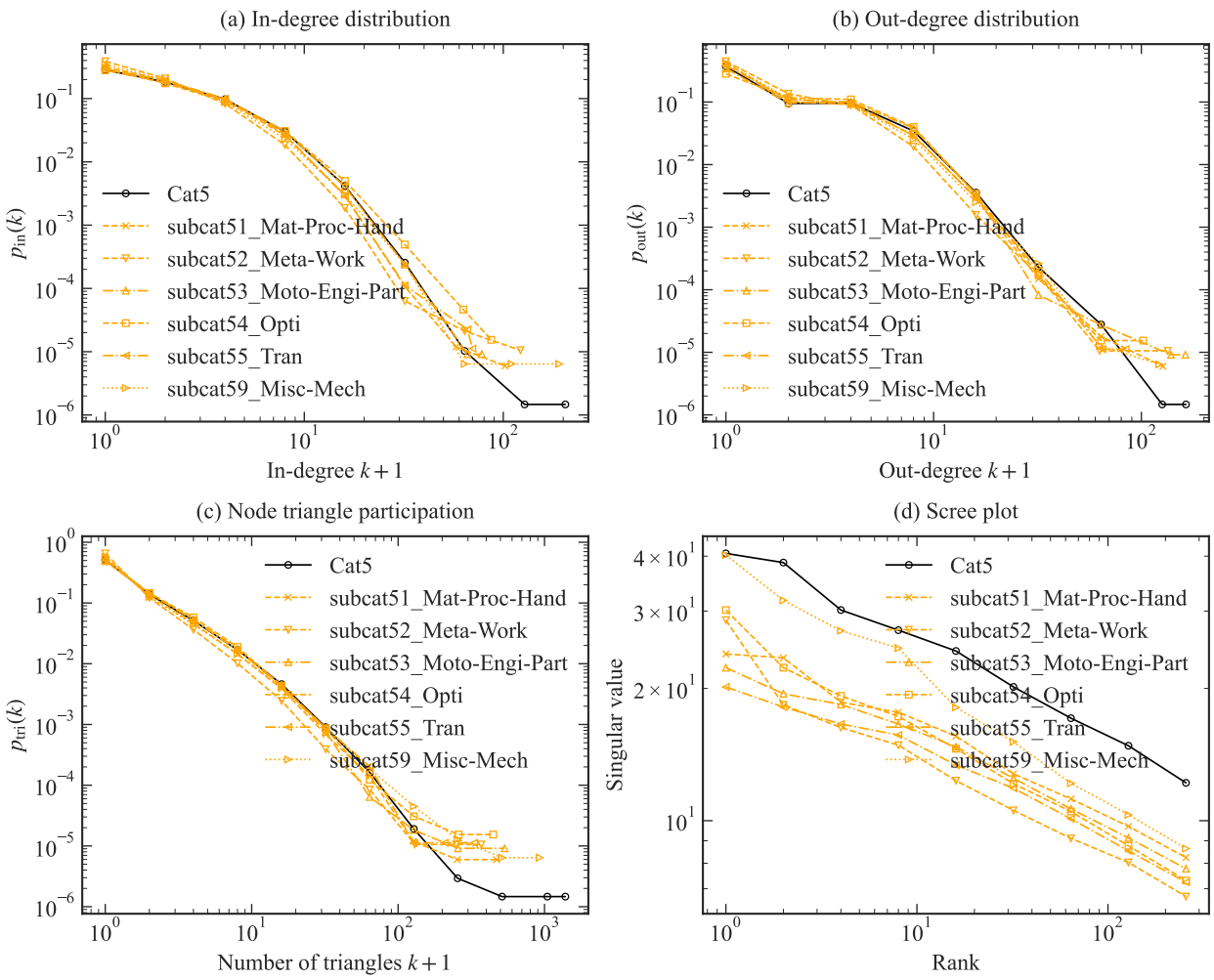


図 4.10. カテゴリ 5 のネットワーク特徴量

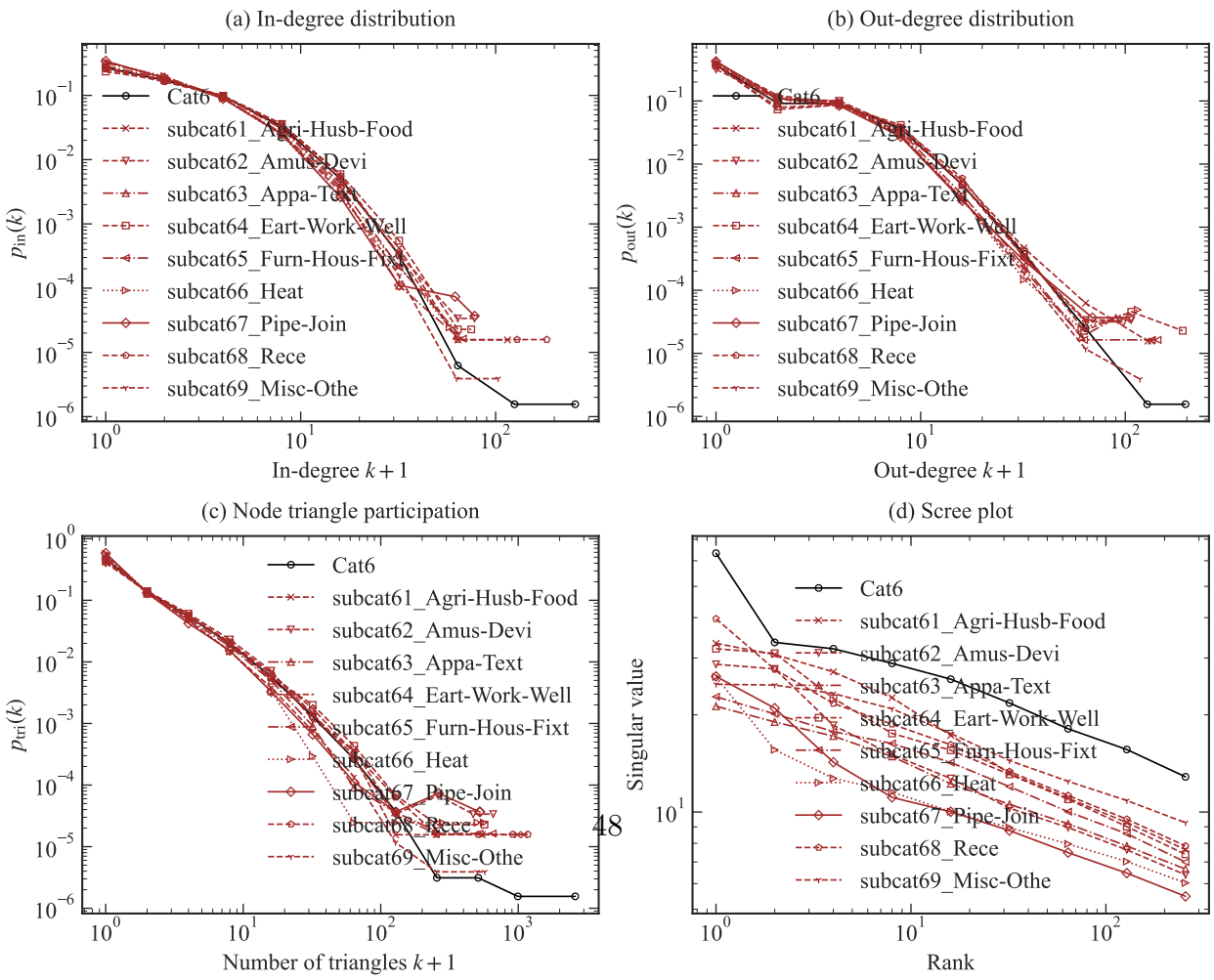


図 4.11. カテゴリ 6 のネットワーク特徴量

4.2　引用ネットワークに対する生成モデルの適合の検証

サブカテゴリ 41 に関連するサブグラフ `cit-Patents-sc41` に着目し，3 章で提案した生成モデルである YN モデル [Yasui and Nakano, 2022b] のモデル適応を確認する．

まずは `WoS-Stat` と同様の手順で f_n, f_c, f_o のパラメータ推定を行う．f_n は時刻 $t \in \{1, 2, ..., T\}$ ごとの文献数 $n(t)$ に対する最小自乗法により，パラメータ $\hat{\mu}_n = 417.015$, $\hat{\sigma}_n = 36.734$, $\hat{\kappa}_n = 1.099 \times 10^8$, and $\hat{\eta}_n = 367.843$ を得た．続いて f_c は時刻差 $s \in \{0, 1, ..., T-1\}$ に対する調整済み時刻差分布 $c(s)$ を用いて最小自乗法により，パラメータ $\hat{\gamma}_c = 2.597$, $\hat{\mu}_c = -0.560$, $\hat{\sigma}_c = 10.324$, $\hat{\kappa}_c = 5.522$ を得た．指数分布に従う確率変数は各ノード v の調整済み出次数 $d^T_{\text{out}}(v)$ に対する最尤推定により， f_o のパラメータ $\hat{\mu}_o = 0.000$, $\hat{\sigma}_o = 4.460$ を得た．f_c と f_o の推定には時刻の範囲 $t \geq 10$ を用いた．

図 4.12 は YN モデルが用いる関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ である．図より問題がないことが確認できる．

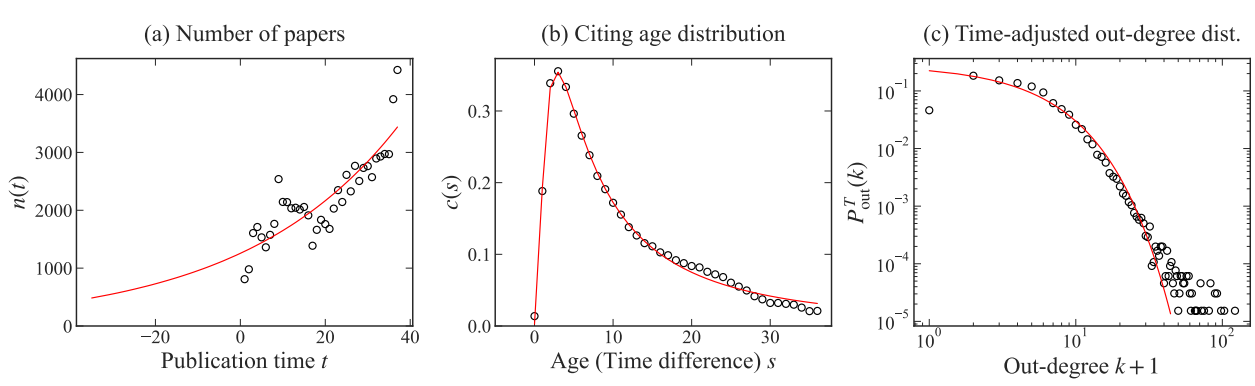


図 4.12. `cit-Patents-sc41` における推定された関数 $\hat{f}_n$, $\hat{f}_c$, $\hat{f}_o$ (赤線) とデータ (黒丸) の比較

図 4.13 は β を 0.3 から 0.7 の範囲を 0.05 ずつ $\beta \in \{0.30, 0.35, ..., 0.70\}$ と変化させて，それぞれ 10 回ずつシミュレーションした結果である．まず (a) 入次数分布，(b) 出次数分布については十分なあてはまりを確認することができる．その一方で (c) 三角形数の分布についてはいずれの β に対しても適合しないことが確認できる．三角形数 $k+1$ とその頻度 $p_{\text{tri}}(k)$ の両対数プロットにおいて，YN モデルが示す形状はアーチ型で，`cit-Patents-sc41` は直線的である．

図 4.14 によると，`WoS-Stat` と同様に β を変化させても (a) 入次数や (b) 出次数のあてはまりは大きく変化しない．一方で (c) 三角形数の分布については，最小値は $\beta = 0.65$ が得られたものの，$0.4 \leq \beta \leq 0.7$ の範囲であまり変化がない．

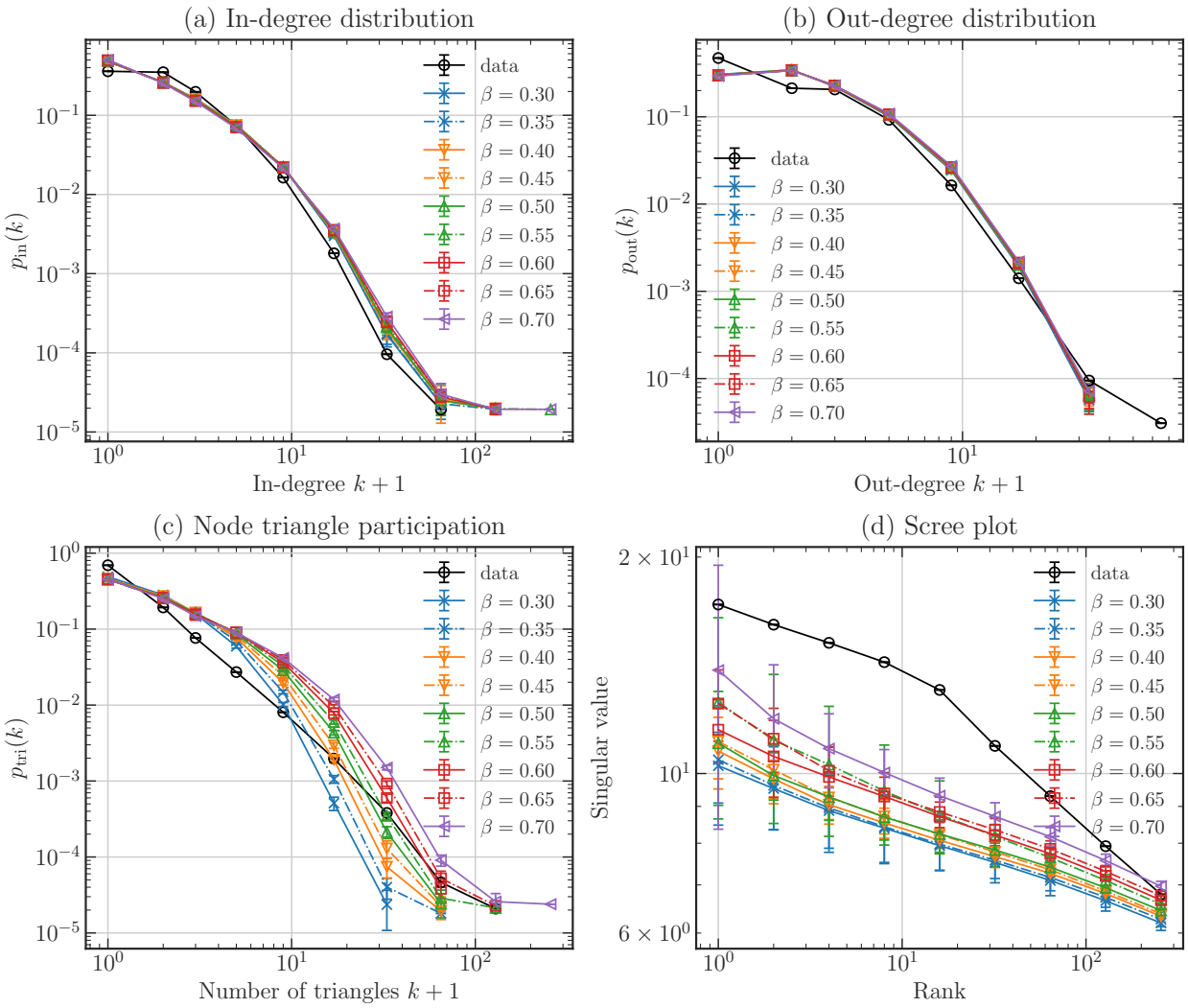


図 4.13. `cit-Patents-sc41` に対する YN モデルのネットワーク特徴量

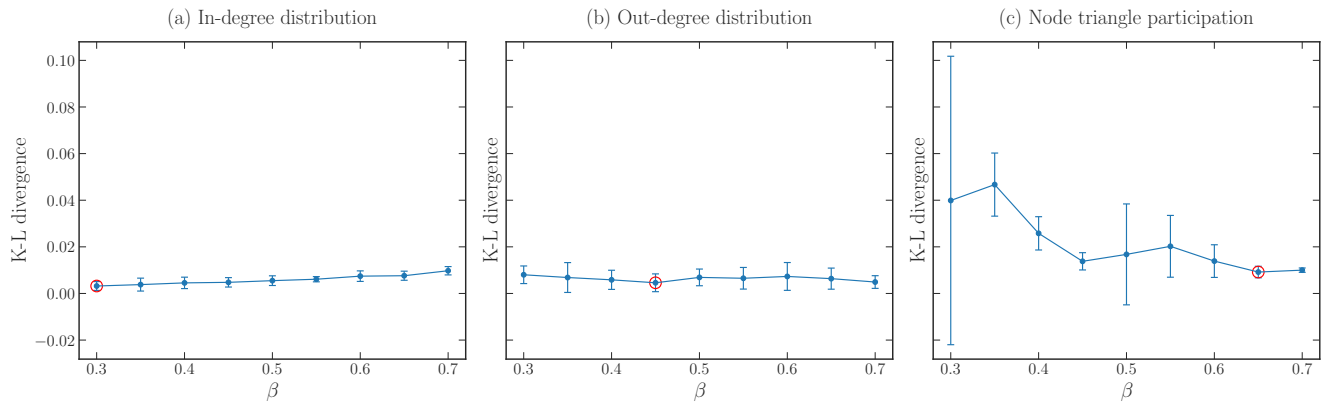


図 4.14. β を変化させたときの，データ `WoS-Stat` に対する YN モデルのシミュレーション結果の Kullback–Leibler 情報量

4.3　各ノードにおける TF の実行割合のモデリング

3 章で提案した YN モデルでは各ノードにおけるエッジ生成のときに TF (Triad Formation) 機構が実行される確率をパラメータ $0 \leq \beta \leq 1$ で与えられた．学術論文の引用ネットワークに対しては β を調整することで三角形数の分布に対する適合を示した．しかしながら前節で示したように特許文献の引用ネットワーク `cit-Patents-sc41` に対してはどのような β を設定したとしても十分な適合を示さない．適合を改善するため，生成モデルの拡張を行う．

4.3.1　各ノードにおける TF の実行回数の予測

各ノードにおける TF の実行確率をモデリングにあたり，まずデータからノード v_i における TF 機構の実行割合を予測を行う．ある対象の引用ネットワーク $G = (V, E)$ の各エッジ $(v, w) \in E$ が，PA (Preferential Attachment) 機構もしくは TF 機構で生成されたと仮定したとき，対象のノード $v_i \in V$ における TF が実行された割合 $p_{\mathrm{TF}}(v_i)$ を予測する．与えられたノード v_i とその隣接ノード集合 $A_{\mathrm{out}}(v_i)$ を PA 機構で選択したノード集合 V_{PA} か TF 機構で選択したノード集合 V_{TF} に分類し，TF 機構の選択確率 $p_{\mathrm{TF}}(v_i) = \frac{|V_{\mathrm{TF}}|}{(|V_{\mathrm{PA}}|-1)+|V_{\mathrm{TF}}|}$ により算出する．ここでノードごとに最初のエッジ生成は PA 機構が実行されることから，分母が $|A_{\mathrm{out}}(v_i)| - 1$ $(= |V_{\mathrm{PA}}| + |V_{\mathrm{TF}}| - 1)$ となることに注意されたい．

Algorithm 2 は近似的に予測を行うアルゴリズムである．このアルゴリズムでは与えられたノード v_i の未分類の隣接ノード集合 $A_{\mathrm{out}}(v_i) \setminus (V_{\mathrm{PA}} \cup V_{\mathrm{TF}})$ に対して，関連するノードにおける最大次数 $|A(v_j) \cap A_{\mathrm{out}}(v_i)|$ となるノード v_j を PA 機構で選択されたとして V_{PA} に追加する．またその隣接ノード集合 $((A(v_j) \cap A_{\mathrm{out}}(v_i)) \setminus V_{\mathrm{PA}})$ を TF 機構で選択されたとして V_{TF} に追加する．このような処理を未分類のノードがなくなるまで繰り返す．

Algorithm 2: ノードにおける TF が実行された割合の予測

Input: 有向グラフ $G = (V, E)$, ノード $v_i \in V$

Result: ノード v_i における TF が実行された割合 $p_{\mathrm{TF}}(v_i)$

Procedure `EstimateProbTF`(G, v_i)**:**

  V_{PA} を $\emptyset$ で初期化

  V_{TF} を $\emptyset$ で初期化

  while $A_{\mathrm{out}}(v_i) = V_{\mathrm{PA}} \cup V_{\mathrm{TF}}$ **do**

    $v_j \leftarrow \underset{v \in A_{\mathrm{out}}(v_i) \setminus (V_{\mathrm{PA}} \cup V_{\mathrm{TF}})}{\arg\max} |A(v) \cap A_{\mathrm{out}}(v_i)|$

    $V_{\mathrm{PA}} \leftarrow V_{\mathrm{PA}} \cup \{v_j\}$

    $V_{\mathrm{TF}} \leftarrow V_{\mathrm{TF}} \cup ((A(v_j) \cap A_{\mathrm{out}}(v_i)) \setminus V_{\mathrm{PA}})$

  if $A_{\mathrm{out}}(v_i) \geq 2$ **then**

    return $\frac{|V_{\mathrm{TF}}|}{(|V_{\mathrm{PA}}|-1)+|V_{\mathrm{TF}}|}$

  else

    return 0

図 4.15, 4.16, 4.17 はそれぞれ `cit-Patents-sc41` に対して Algorithm 2 がノード v_{23246}, v_{24199}, v_{22702} （図中，青四角のノード）の隣接ノード集合が PA 機構（図中，赤丸のノード）と TF 機構（図中，黄五角形のノード）のいずれかで選択したかを表している．いずれも PA で選択されたノードに比べて，その周辺の TF で選択されたノードは次数が小さいことが確認できる．

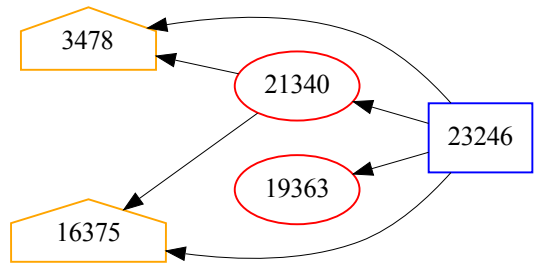


図 4.15. ノード v_{23246} における TF 機構の実行割合 $p_{\mathrm{TF}}(v_{23246}) = \frac{2}{4-1} = 0.666$

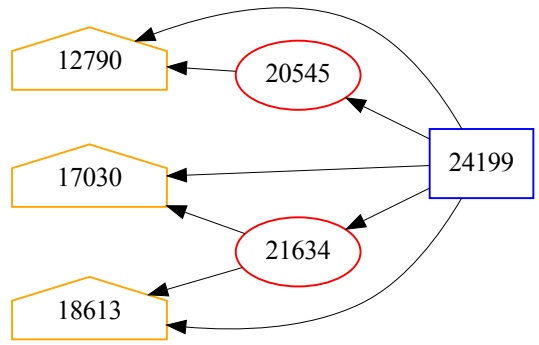


図 4.16. ノード v_{24199} における TF 機構の実行割合 $p_{\mathrm{TF}}(v_{24199}) = \frac{3}{5-1} = 0.750$

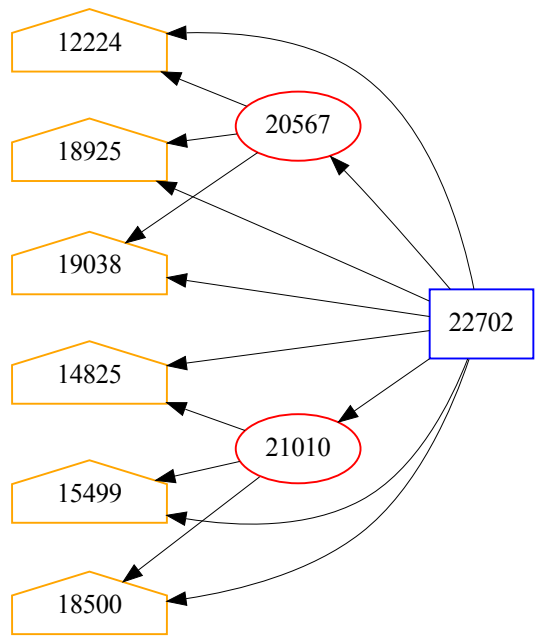
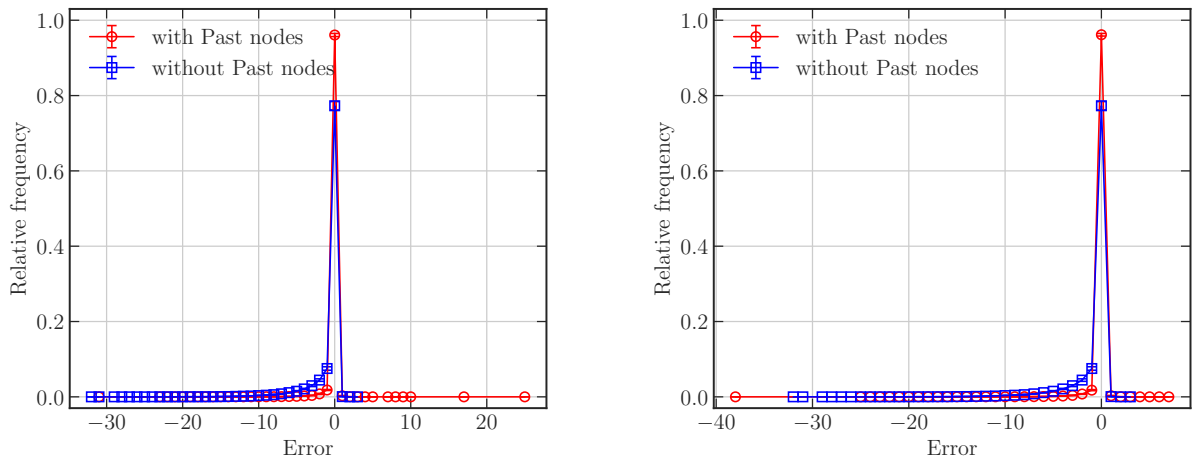


図 4.17. ノード v_{22702} における TF 機構の実行割合 $p_{\mathrm{TF}}(v_{22702}) = \frac{6}{8-1} = 0.857$

図 4.18a と図 4.18b は，それぞれ `arXiv-HepTh` と `arXiv-HepPh` に対し YN モデルで生成したネットワークを用いて，Algorithm 2 の精度を検証した結果である．Algorithm 2 により各ノードにおける TF 機構が実行された回数 k_{gen} とその予測回数 k_{pred} の誤差 $\mathrm{Error} = k_{\mathrm{pred}} - k_{\mathrm{gen}}$ をプロットした．その際，時刻の範囲外 $(t \leq 0)$ となるノードを残したとき (with past nodes) と除外したとき (without past nodes) の結果を比較した．それぞれ 10 回ずつのシミュレーションのときの一致率は，`arXiv-HepTh` に対して範囲外のノードを残したときは $96.077 \pm 0.417\%$ で，除外したときは $77.293 \pm 1.107\%$ であった．また `arXiv-HepPh` に対しては範囲外のノードを残したときは

96.153 ± 0.350% で，除外したときは 77.616 ± 0.468% であった．このように各ノードから張られるエッジについて，どの時点でも T 期さかのぼることができる場合は 95% ほどの精度で，各エッジが PA と TF どちらで生成されたのか予測することは可能である．しかしながらある時刻の範囲に含まれるノードを対象とすると精度は 75% ほどに低下する．さらに誤差は負値で多く観測されることから，小さめに推定される傾向があるといえる．



(a) `arXiv-HepTh` **(b)** `arXiv-HepPh`

図 4.18. 生成時に TF 機構が実行された回数 k_{gen} と，その予測回数 k_{pred} の差 $\mathrm{Error} = k_{\mathrm{pred}} - k_{\mathrm{gen}}$ の頻度．時刻の範囲外のノードを残す場合と除外する場合の比較．

図 4.19 は `cit-Patents-sc41` に対し Algorithm 2 を適用して，得られた p_{TF} の分布を示したものである．左図 (a) はヒストグラム全体を，右図 (b) は $0.001 \leq p_{\mathrm{TF}}$ に限定したヒストグラムをそれぞれプロットした結果である．図より 0 や 1 となる割合が大きく，それぞれ 64.5 % のノードが 0 に，3.7 % のノードが 1 となる．なおヒストグラムとして可視化する対象ノードは出次数が 2 以上のみとしたが，ノードの出次数が 2 未満である場合は必ず 0 となることから，出次数が 2 未満を含む全体での 0 の割合は 77.5% にも及ぶ．

図 4.20 は `cit-Patents-sc41` に対して，ノード $v \in V$ の出次数 $d_{\mathrm{out}}(v)$ がある範囲 $[4, 8)$, $[8, 16)$, $[16, 32)$, $[32, 64)$ に該当するときの TF の実行割合 $p_{\mathrm{TF}}(v)$ をヒストグラムで可視化した．なお $d_{\mathrm{out}}(v) = 2$ は 0 もしくは 1 のみを，$d_{\mathrm{out}}(v) = 3$ は $\{0, 0.5, 1\}$ のみを取るため，可視化から除外している．図より, 出次数がある範囲 $d_{\mathrm{out}}(v) \geq 16$ となる範囲で分布の形状が安定することが確認できる．また $d_{\mathrm{out}}(v) < 16$ では 0 の割合が多く，$d_{\mathrm{out}}(v) \geq 16$ では 1 の割合が多い．次数が大きくなると TF の実行確率 p_{TF} の分布は増大する傾向にあることを確認できる．

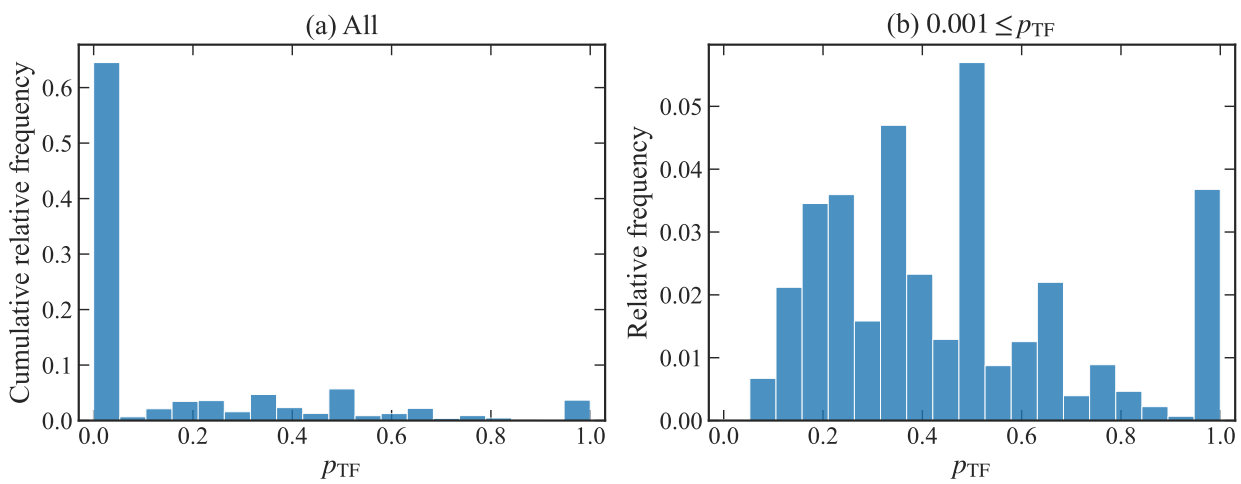


図 4.19. `cit-Patents-sc41` における TF の実行割合 p_{TF} の分布

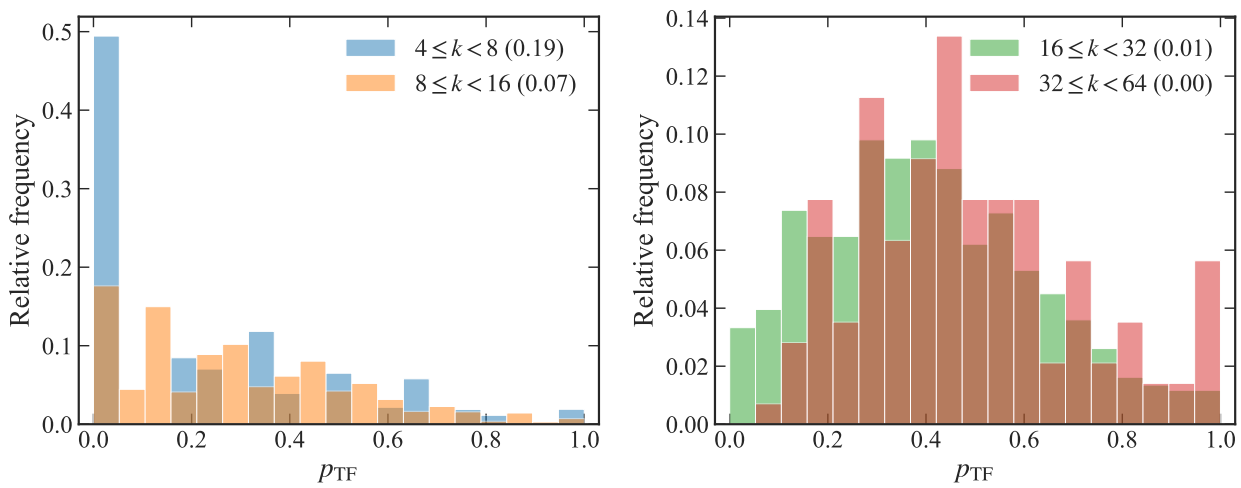


図 4.20. `cit-Patents-sc41` におけるノードの出次数 k ごとの TF の実行確率 p_{TF} の分布

4.4　特許文献の引用ネットワークに対する生成モデル

TF 機構の実行割合の分布が確率分布に従うと仮定してモデル化を行う．前節での観察により，アルゴリズムにより予測された TF 機構の実行割合は 0 や 1 が多く含み，小さめに推定され，また次数に応じて大きくなるなどの性質を有する．それらを修正するための補正を行う．

4.4.1　TF 機構が実行する割合のモデル化

TF 機構が実行する割合 p_{TF} はゼロワン過剰ベータ分布 [Ospina and Ferrari, 2010] に従う確率変数と仮定する．ゼロワン過剰ベータ分布の確率密度関数 (PDF) はガンマ関数 Γ を用いて

$$f_b(x \mid \pi_0, \pi_1, a, b) = \begin{cases} \pi_0 & \text{if } x = 0 \\ \pi_1 & \text{if } x = 1 \\ (1 - \pi_0 - \pi_1)\dfrac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1}(1-x)^{b-1} & \text{if } 0 < x < 1 \end{cases} \tag{4.1}$$

で与えられる．さらに f_b に従う確率変数 x に対して，対象ノードの出次数 k とパラメータ k_U を用いて

$$f_\beta(k, k_U) = \begin{cases} 1 & \text{if } k_U \leq k \\ x & \text{otherwise} \end{cases} \tag{4.2}$$

と構成する．アルゴリズム 2 は p_{TF} を小さめに予測し，また p_{TF} は次数に応じて大きくなる性質を有していることから，f_β では出次数 k が k_U 以上の場合に 1 と補正する．

4.4.2　生成プロセスのモデル化

関数 f_n, f_c, f_o, f_β とパラメータ k_U にもとづき PA 機構と TF 機構を組み合わせた成長モデルの生成プロセスについて構成する．

時刻 t でサイズが $\lfloor f_n(t) + \epsilon_n(t) \rfloor$ となるノード集合がネットワークに追加する．追加されたノードは f_o にもとづいて生成された x の整数部分となる出次数 $k = \lfloor x \rfloor$ や，$f_\beta(k, k_U)$ に従い生成された TF 機構の実行割合 β をもつ．PA 機構の実行確率は $1 - \beta$ となる．PA 機構もしくは TF 機構を用いて，k 回のエッジ生成を行う．

ノード v_i がネットワークに追加されたとき，PA 機構はすでにネットワークに存在するノード集合からノード $v_j \in V$ は確率 $\Pi_{\mathrm{PA}}(v_i, v_j)$ (式 (3.1)) で選択し，エッジ (v_i, v_j) をネットワークに追加する．このとき $Im(v_j)$ は v_j の重要度，$f_c(\tau(v_i) - \tau(v_j))$ は時刻差 $\tau(v_i) - \tau(v_j)$ における引用率の経年変化である．TF 機構はノード集合 C の各ノードの隣接ノード集合から $v_k \in \bigcup_{u \in C} A(u)$ を確率 $\Pi_{\mathrm{TF}}(v_i, v_k)$ (式 (3.2)) で選択し，エッジ (v_i, v_k) をネットワークに追加する．ここで C はノード v_i がネットワークに追加されてから，PA 機構によって選択されたノード集合である．ノード v の重要度 $Im(v)$ は $d_{\mathrm{in}}(v) + 1$ で近似する．

4.4.3　3 章で提案したモデルとの比較

前節で説明した生成モデルは 3 章で提案した YN モデルをベースラインとして，2 種類の観点で拡張を行っている．まず YN モデルでは， TF 機構で選択される候補を直前の PA で選択されたノードの隣接ノード集合で設定しているのに対し，拡張 (1) では候補を基準となるノードが追加されてから PA で選択されたノードの隣接ノード全てに拡大した．また YN モデルでは， TF 機構の実行割合は定数パラメータ $0 \leq \beta \leq 1$ で指定するのに対し，拡張 (2) では f_β でモデル化している．

それぞれ YN モデルに対して，拡張 (1) のみを適用した「YN モデル + 拡張 (1)」，拡張 (2) のみを適用した「YN モデル + 拡張 (2)」，拡張 (1) と (2) を適用した「YN モデル + 拡張 (1) + 拡張 (2)」として，比較を行う．

4.5　関数群の推定とシミュレーション

4.5.1　パラメータの推定

`cit-Patents-sc41` に対する f_n, f_c, f_o, f_b のパラメータ推定について説明する．f_n, f_c, f_o については，すでに図 4.12 で説明している．各ノード v ごとに予測された TF の実行割合 $p_{\mathrm{TF}}(v)$ のうち，時刻 $t \geq 20$ かつ $d_{\mathrm{out}} \geq 2$ を対象に最尤推定により f_b のパラメータ $\hat{\pi_0} = 0.555$, $\hat{\pi_1} = 0.046$, $\hat{a} = 2.467$, $\hat{b} = 3.760$ を得た．推定と生成には Python の SciPy パッケージ [Virtanen et al., 2020] と，R の VGAM パッケージ [Yee, 2015] を用いた．

図 4.21 は `cit-Patents-sc41` における TF の実行割合 p_{TF} と推定された関数 $\hat{f_b}$ を累積分布関数で比較したものである．図より十分に適合していることを確認できる．図 4.22 は `cit-Patents-sc41` における TF の実行割合 p_{TF} と，推定されたパラメータに従い生成した TF の実行割合列をヒストグラムで比較したものである．左図は全体を，右図を 0 と 1 を除外したものだが，いずれも十分に適合していることを確認できる．

4.5.2　ネットワーク生成のためのシミュレーション

まずノード集合 V' とエッジ集合 E' を空集合 $\emptyset$ で初期化する．その後, 時刻 t を $-T+1$, $-T+2$, ..., T と 1 ずつ変化させて，後続の処理を実行する．まず時点 t において，アルゴリズムはサイズが $\lfloor f_n(t) + \epsilon_n(t) \rfloor$ となるノード集合を V' に追加する．そのとき $t \geq 0$ であれば追加されたノード $v_i \in U$ に対して PA 機構や TF 機構によるエッジ生成を行い，$t \leq 0$ であれば何もしない．v_i が生成するエッジの数 k は f_o にもとづいて生成した乱数 x の整数部分 $k = \lfloor x \rfloor$ で決定する．また TF 機構が実行される割合は $f_\beta(k, k_U)$ にもとづいて生成された β で設定する．エッジ生成の 1 回目は PA 機構が実行され，2 回目以降は PA 機構を確率 $1-\beta$ で，TF 機構を確率 β で実行する．

PA 機構は式 (3.1) にもとづいてノード v_j を選択し，エッジ (v_i, v_j) を E' に追加する．まず時刻差 $s \in \{0, 1, \ldots, T-1\}$ を $f_c(s)$ に比例する確率で決定し，その後，ノード $v_j \in$

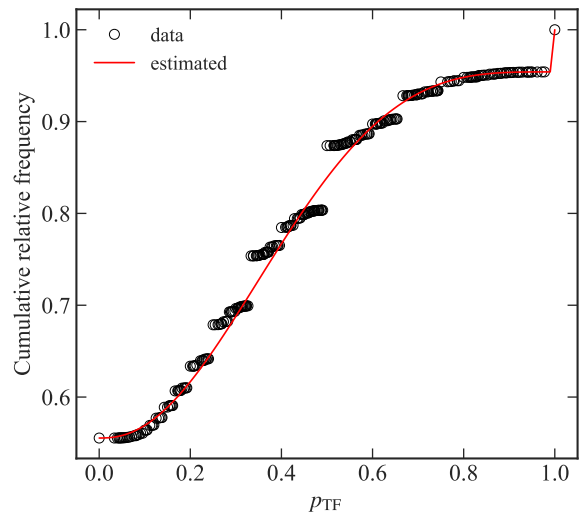


図 4.21. `cit-Patents-sc41` における推定された関数 $\hat{f}_b$ (赤線) とデータ (黒丸) の比較

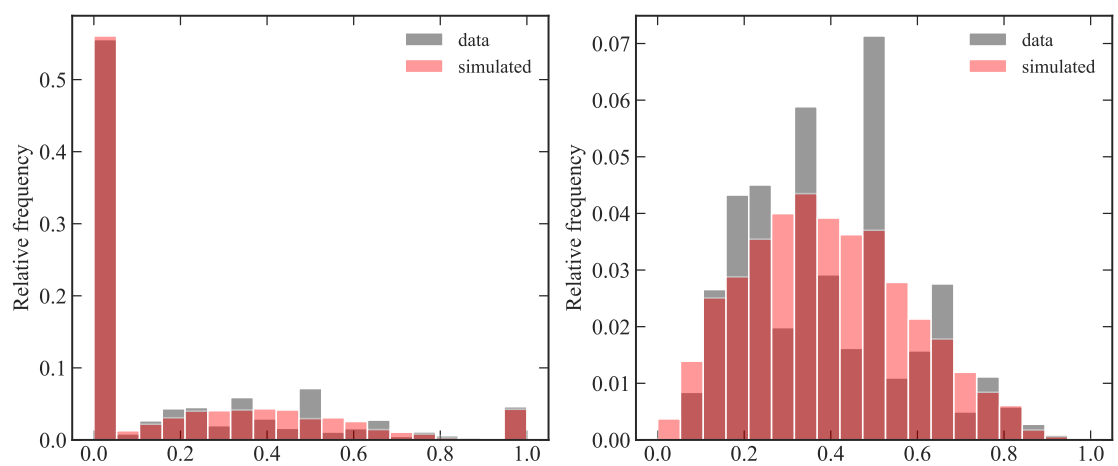


図 4.22. `cit-Patents-sc41` における関数 $\hat{f}_b$ からシミュレーションされた TF の実行割合 (赤線) とデータ (黒丸) の比較．左図は全体を，右図は $0.001 \leq p_{\mathrm{TF}} \leq 0.999$ のみを比較．

$\{v \mid v \in V',\ \tau(v_i) - \tau(v) = s\}$ を $d_{\mathrm{in}}(v_j) + 1$ に比例する確率で選択する．

一方, TF 機構では，式 (3.2) にもとづいて周辺ノード集合 $W(s)$ からノード v_k を選択し，ノード v_i とのエッジ (v_i, v_k) を E' に追加する．ここで C はノード v_i が追加されたときに空集合 $\emptyset$ で初期化され，PA 機構により選択された v_j からなるノード集合となる．$W(s)$ は s ごとに

$$W(s) = \left\{ v \;\middle|\; v \in \bigcup_{u \in C} A(u),\ \tau(v_i) - \tau(v) = s \right\} \setminus \{v_i\} \tag{4.3}$$

と構成する．その後，時刻差 s を $f_c(s)$ に比例する確率で選択し，$v_k \in W(s)$ を $d_{\mathrm{in}}(v_k) + 1$ に比例する確率で選択する．ここで $W(s)$ が空集合になる s に対しては $f_c(s) = 0$ とする．全ての $W(s)$ が空集合である場合，TF 機構の代わりに PA 機構を実行する．

最後に, V' や E' のうち，時刻の範囲外となるノードやエッジを除外した V, E を出力する.

$$V = \{v \mid v \in V',\ 1 \leq \tau(v) \leq T\} \tag{4.4}$$

$$E = \{(v_i, v_j) \mid v_i, v_j \in V,\ (v_i, v_j) \in E'\} \tag{4.5}$$

以上の手順を Algorithm 3 としてまとめた.

Algorithm 3: GenerateCitationNetworkWithExtensions

Input: 各時刻 $t \in \{-T+1, ..., T\}$ の文献数 $f_n(t)$, 各時刻差 $s \in \{0, ..., T-1\}$ の時刻分布 $f_c(s)$, 時刻調整された出次数分布 f_o, TF 機構の実行割合の分布 f_β, パラメータ k_U

Result: 生成されたグラフ (V, E)

Procedure `TriadFormation`(f_c, v_i, C):
  時刻差 $s \in \{0, 1, ..., T-1\}$ ごとに
  $W(s) = \left\{v \;\middle|\; v \in \bigcup_{u \in C} A(u),\ \tau(v_i) - \tau(v) = s\right\} \setminus \{v_i\}$ を構成　　// 拡張（1）
  時刻差 $s \in \{0, 1, ..., T-1\}$ を $f_c(s)$ に比例する確率で選択．ただし $W(s)$ が $\emptyset$ である場合は確率は 0 とする．
  ノード $v_k \in W(v_i, v_j, s)$ を $d_{\text{in}}(v_k) + 1$ に比例する確率で選択
  return v_k

Procedure `GenerateCitationNetworkWithExtensions`(T, f_n, f_c, f_o, f_β, k_U):
  (V', E') を $(\emptyset, \emptyset)$ で初期化
  for $t_i \leftarrow -T+1$ **to** T **do**
    $U \leftarrow \{v_{|V'|+i} \mid i \in \{1, 2, .., \lfloor f_n(t_i) \rfloor\}$
    $V' \leftarrow V' \cup U$
    if $t_i < 0$ **then**
      contitnue
    foreach $v_i \in U$ **do**
      $v_j \leftarrow \emptyset$
      f_o に従う乱数 r を生成し, $k \leftarrow \lfloor r \rfloor$ とする
      $f_\beta(k, k_U)$ に従う乱数 β を生成する　　// 拡張（2）
      $C \leftarrow \emptyset$　　// 拡張（1）
      for 1 **to** k **do**
        if $C \neq \emptyset$ **or** `Random(0,1)` $< \beta$ **then**
          $v_k \leftarrow$ `TriadFormation`(f_c, v_i, C)　　// 拡張（1）
          if $v_k \neq \emptyset$ **then**
            $E' \leftarrow E' \cup \{(v_i, v_k)\}$
            contitnue
        $v_j \leftarrow$ `PreferentialAttachment`(V', f_c, v_i)
        $C \leftarrow C \cup \{v_j\}$　　// 拡張（1）
        $E' \leftarrow E' \cup \{(v_i, v_j)\}$
  $V \leftarrow \{v \mid v \in V',\ 1 \leq \tau(v) \leq T\}$
  $E \leftarrow \{(v_i, v_j) \mid v_i, v_j \in V,\ (v_i, v_j) \in E'\}$
  return (V, E)

4.6　特許文献の引用ネットワークに対するシミュレーション

図 4.23 は `cit-Patents-sc41` に対して，モデルと拡張 (1) と (2)，シミュレーションで決定するパラメータ β, k_U の組合せに対して K-L 情報量により適合を確認する．

YN は 3 章で提案した生成モデルでベースラインとなる．YN+Ex(1) は YN に拡張 (1) である TF 機構の候補を拡大した拡張を適用したモデルである．YN+Ex(2) は YN に拡張 (2) である各ノードにおける TF 機構の実行確率を f_β でモデル化した拡張をしたモデルである．YN+Ex(1)+(2) は拡張 (1) と (2) を両方とも適用したモデルである．

まず `WoS-Stat` と同様に β や k_U を変化させても (a) 入次数分布や (b) 出次数分布は大きく変化しない．そこで (c) 三角形数の分布に焦点を当てて，各パラメータが適合にどのような影響を与えるか確認する．YN+Ex(1) は $\beta = 0.55$ で適合の改善を実現でき，YN における不安定な性質が一定程度改善されており，拡張の効果が高いことが確認できる．さらに YN+Ex(2) や YN+Ex(1)+(2) は β の代わりに k_U を変化させる．まず f_β における補正は効果が高いことが確認できる．補正しない場合 $k_U = \infty$ ではベースライン YN よりも性能が悪化する．一方で, $k_U = 16$ については比較対象の中で適合が最高となる結果を得た．$k_U = 16$ は出次数が上位 1.323% のノードが該当する．

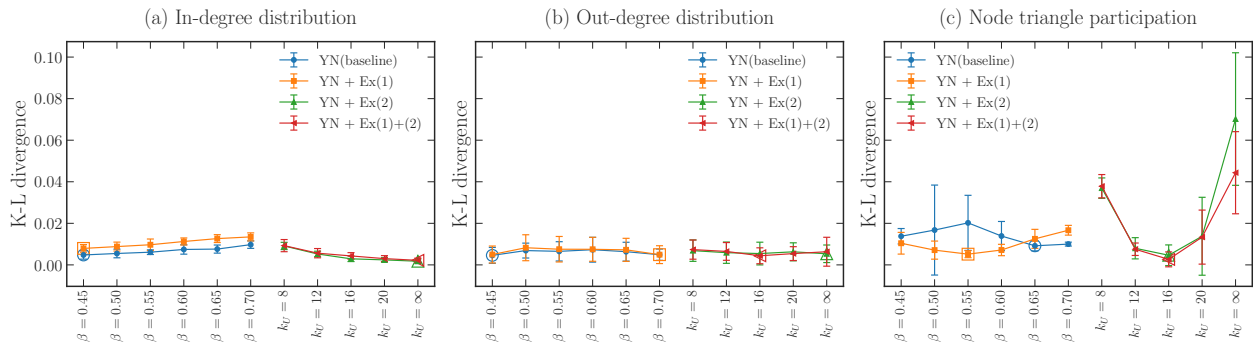


図 4.23. データ `cit-Patents-sc41` に対するモデルの拡張と，パラメータ β や k_U を変化させたシミュレーション結果の K–L 情報量

図 4.24, 4.25, 4.26 は，図 4.23 で比較したモデルのうち適合が高いものをいくつか選出し，ネットワーク特徴量で比較したものである．

まず図 4.24 では (a) 入次数分布 と (b) 出次数分布についてはいずれも適合しており，拡張 (1), (2) による影響は小さい．(c) 三角形数の分布については YN はアーチ型の形状であるため適合しない．YN+Ex(1) を用いると若干その傾向は改善されている．それらに対し YN+Ex(2) もしくは YN+Ex(1)+(2) に $k_U = 16$ を設定すると，大幅に改善することが確認できる．(d) Scree plot については YN+Ex(1) が最も適合しており，YN+Ex(1)+(2) と YN が同等，YN+Ex(2) へと続く．

図 4.25 は拡張 (1) の効果に着目し，(c) 三角形数の分布と (d) scree plot で比較したものである．(c) 三角形数の分布については YN に対して，YN+Ex(1) は三角形数が大きいところでの改善が確認できる．(d) Scree plot についても YN+Ex(1) の改善を確認できた．

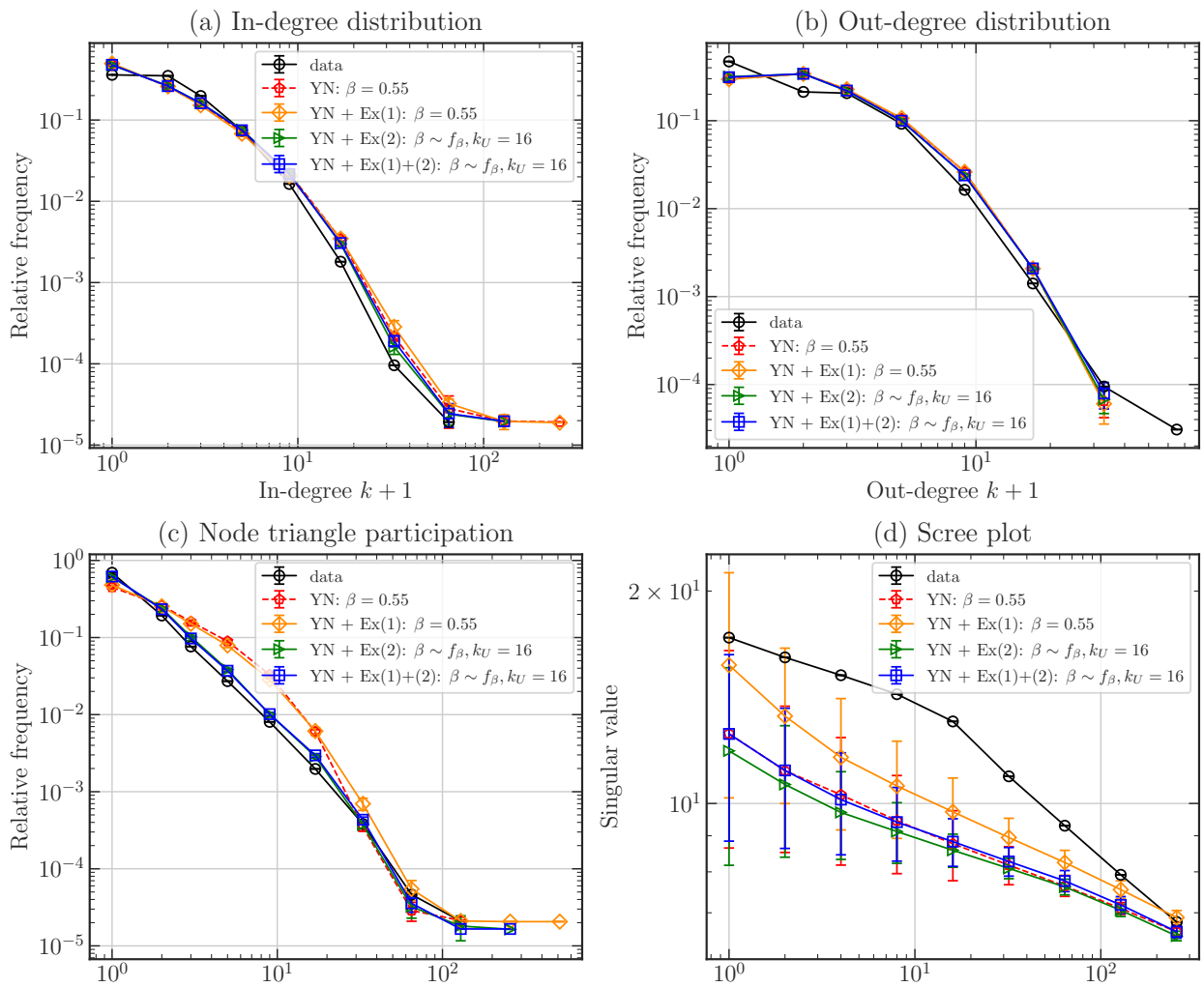


図 4.24. 引用ネットワーク `cit-Patents-sc41` に対するモデル適合の比較

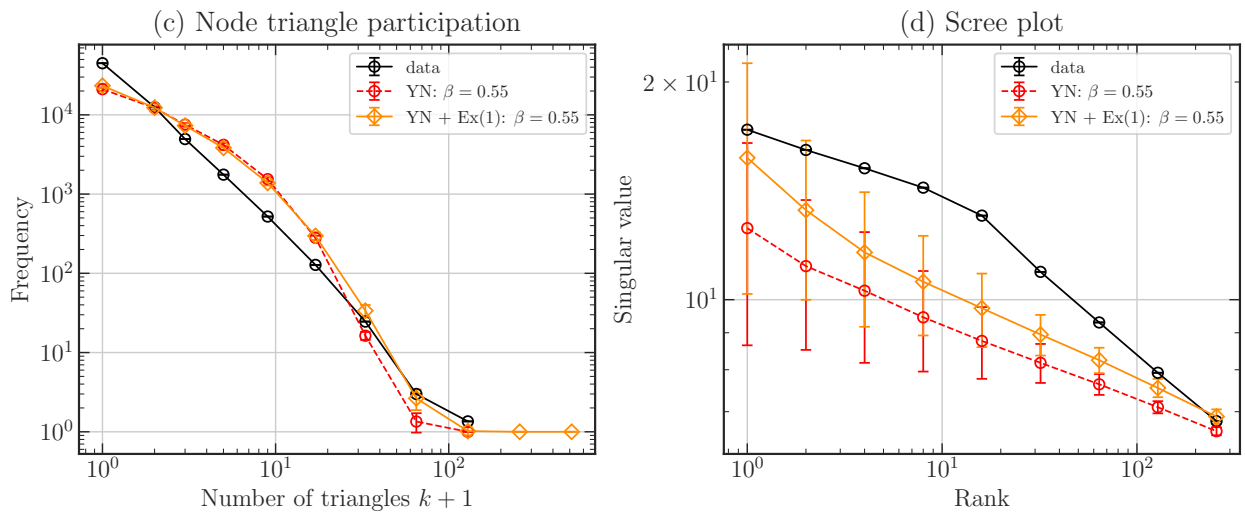


図 4.25. 引用ネットワーク `cit-Patents-sc41` に対する拡張 (1) に着目したモデル適合の比較

図 4.26 は拡張 (2) の効果に着目し，(c) 三角形数の分布と (d) scree plot で比較したものである．(c) 三角形数の分布については YN+Ex(2) と YN+Ex(1)+(2) どちらにおいても，k_U に適合が依存することが確認できた．$k_U = 16$ など適切に設定すると高い適合を得られる一方で，$k_U = \infty$ では三角形数が大きいところでの適合の度合いが低下する．しかしながら補正の効果のない，大部分を占める三角形数の小さいところでは YN+Ex(2) と YN+Ex(1)+(2) どちらのモデルも十分な適合であったから，拡張 (2) の効果を確認することができる．(d) Scree plot についても三角形数の分布と同様の傾向を示しつつ，僅かだが YN+Ex(1)+(2) がより高い適合を示している．

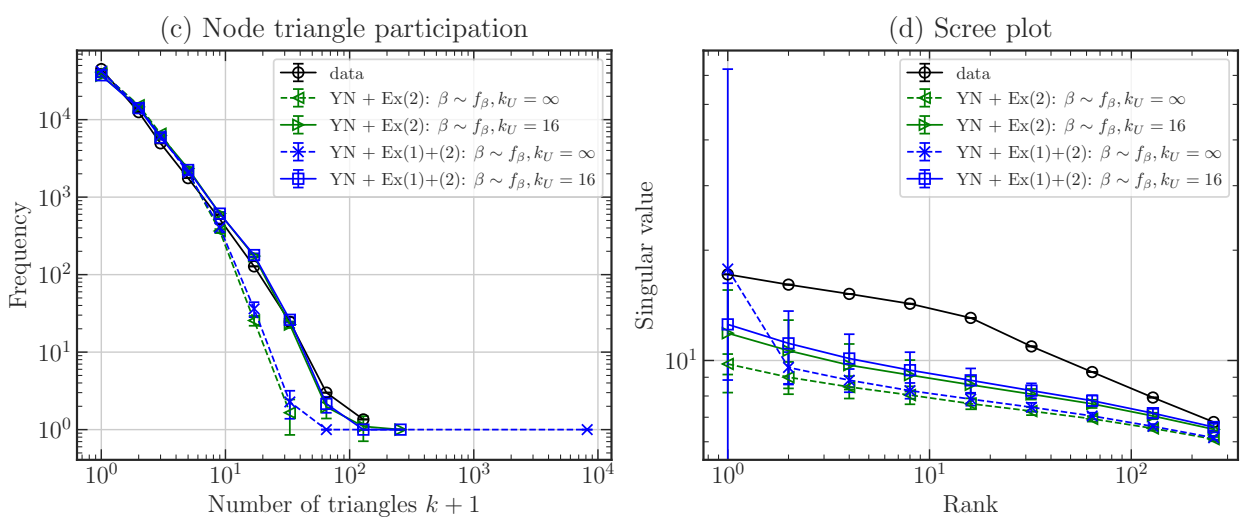


図 4.26. 引用ネットワーク `cit-Patents-sc41` に対する拡張 (2) に着目したモデル適合の比較

4.7　学術文献の引用ネットワークに対するシミュレーション

拡張されたモデルの一般性を示すために，学術論文の引用ネットワーク `WoS-Stat` を用いての適合を検証する．まず 3 章で推定したパラメータに加えて，各ノード v ごとに予測された TF の実行割合 $p_{\mathrm{TF}}(v)$ のうち，時刻 $t \geq 20$ かつ $d_{\mathrm{out}} \geq 2$ を対象に，最尤推定により f_b のパラメータ $\hat{\pi_0} = 0.153$, $\hat{\pi_1} = 0.101$, $\hat{a} = 4.139$, $\hat{b} = 2.819$ を得た．

図 4.27 は `WoS-Stat` における TF の実行割合 p_{TF} と推定された関数 $\hat{f}_b$ を累積分布関数で比較したものである．図より十分に適合していることを確認できる．図 4.28 は `WoS-Stat` における TF の実行割合 p_{TF} と，推定されたパラメータに従い生成した TF の実行割合列をヒストグラムで比較したものである．左図は全体を，右図を 0 と 1 を除外したものだが，いずれも十分に適合していることを確認できる．

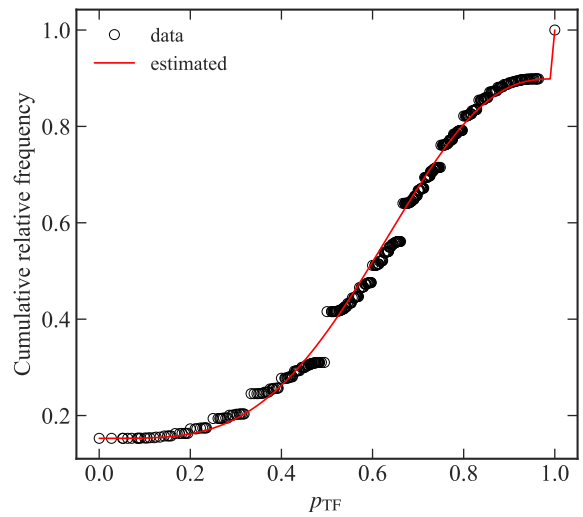


図 4.27. `WoS-Stat` における推定された関数 $\hat{f}_b$ (赤線) とデータ (黒丸) の比較

図 4.29 は `WoS-Stat` に対して，モデルと拡張 (1) と (2)，シミュレーションで決定するパラメータ β, k_U の組合せに対して K-L 情報量により適合をまとめたものである．`WoS-Stat` に対しては拡張 (1) が効果が薄いと判断したため，比較の対象から YN+Ex(1) モデルを除外した．図 4.23 の結果と同様に，β や k_U を変化させても (a) 入次数分布や (b) 出次数分布は大きく変化しない．また (c) 三角形数の分布についても，YN+Ex(2) や YN+Ex(1)+(2) について補正しない場合 $k_U = \infty$ ではベースライン YN よりも悪化する．一方で, $k_U = 12$ と設定すると比較対象の中で適合が最高となる結果を得た．$k_U = 12$ は出次数が上位 13.968% のノードが該当する．

図 4.30, 4.31 は，図 4.29 で比較したモデルのうち適合が高いものをいくつか選出し，ネットワーク特徴量で比較したものである．図 4.30 において，いずれのモデルも (a) 入次数分布 と (b) 出次数分布に適合しており，拡張による影響は小さい．(c) 三角形数の分布については YN ですでに適合しているが，YN+Ex(2) もしくは YN+Ex(1)+(2) に $k_U = 12$ を設定すると，改善が確認できる．(d) Scree plot については YN+Ex(1)+(2) が最も適合しており，YN+Ex(1)+(2)， YN へと続く．

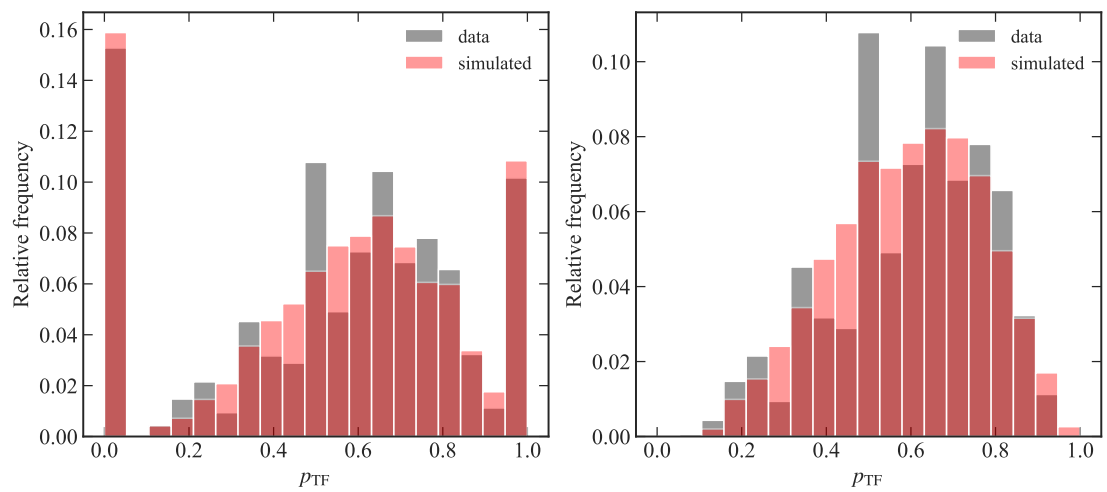


図 4.28. `WoS-Stat` における関数 $\hat{f}_b$ からシミュレーションされた TF の実行割合 (赤線) とデータ (黒丸) の比較．左図は全体を，右図は $0.001 \leq p_{\mathrm{TF}} \leq 0.999$ のみを比較．

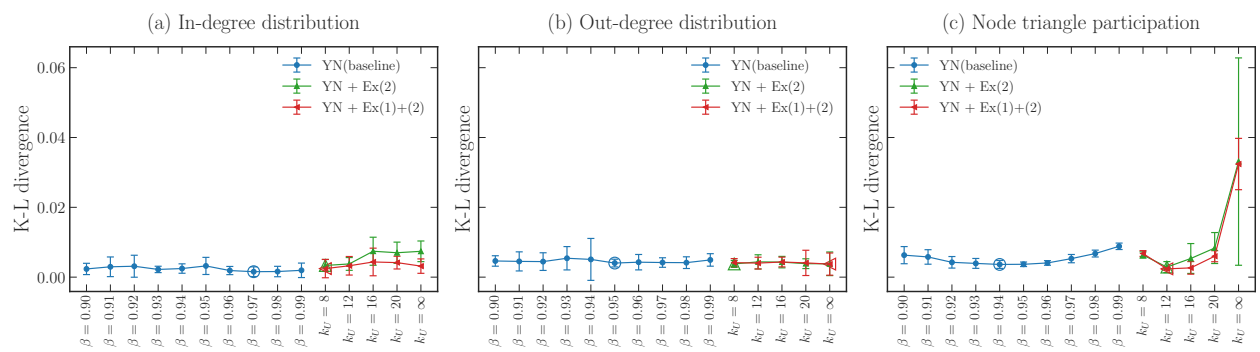


図 4.29. データ `WoS-Stat` に対する β や k_U を変化させたシミュレーション結果の K–L 情報量

図 4.31 は拡張 (2) の効果に着目し，(c) 三角形数の分布と (d) scree plot で比較したものである．(c) 三角形数の分布については $k_U = 12$ など適切に設定すると高い適合を得られる一方で，$k_U = \infty$ では三角形数が大きいところでの適合の度合いが低下する．大部分を占める三角形数の小さいところでは YN+Ex(2) と YN+Ex(1)+(2) どちらのモデルも十分な適合していることから，拡張 (2) の効果を確認できる．(d) Scree plot についても三角形数の分布と同様の傾向を示した．

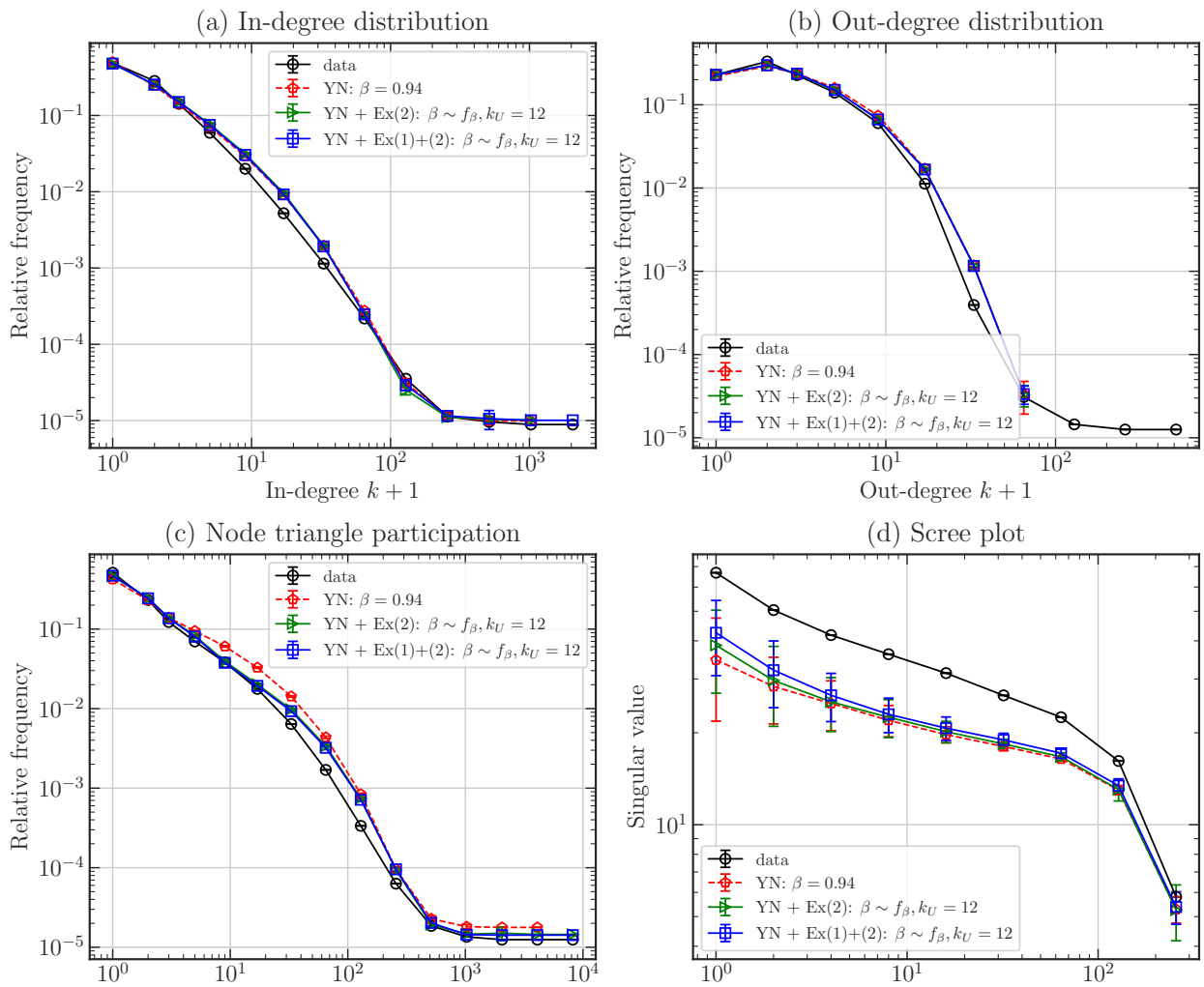


図 4.30. 引用ネットワーク `WoS-Stat` に対するモデル適合の比較

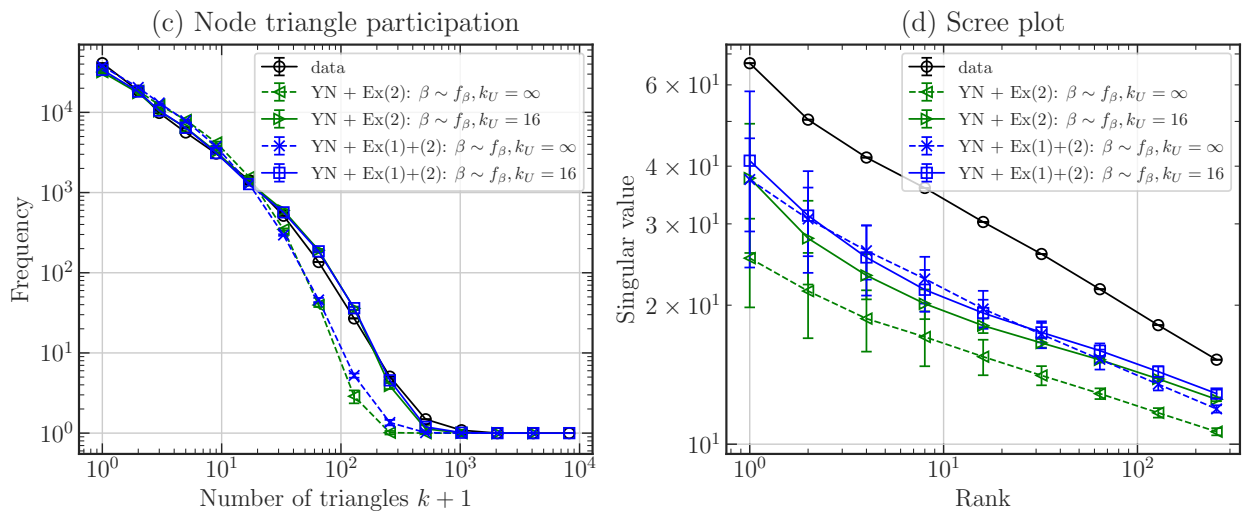


図 4.31. 引用ネットワーク `WoS-Stat` に対する拡張 (2) に着目したモデル適合の比較

4.8　モデル拡張とシミュレーション結果の解釈

本章では特許文献から構築された引用ネットワークに対する生成モデルを提案した．対象の引用ネットワークはカテゴリとサブカテゴリからなる階層的なクラスタ構造をもつため，前章で扱った生成モデルは均一の性質をもつ引用構造を前提にするため，対応することができない．そこでサブカテゴリ 41 に着目し，サブカテゴリ内の引用構造を対象とした生成モデルについて議論を行った．

前章で扱った提案モデルで必要な関数のあてはまりは問題ないことを確認できたものの，そのままではシミュレーションで得られるネットワーク構造はデータと十分に類似な構造を有しているとはいえない．そこで該当モデルに対して，拡張 (1), (2) を行い，新たなモデルを提案する．

拡張 (1) は TF 機構で選択されるノードの候補を，直前だけでなくそれまでの PA 機構の隣接ノード集合への拡大である．シンプルな拡張であるものの，適合に対する改善を K-L 情報量を用いたモデル比較やネットワーク特徴量のプロットで確認できる．あるノードからのエッジ生成を行う際に TF 機構により選択されるノードの数に対し，候補が十分に確保できなかったためであると推測できる．

拡張 (2) では TF 機構の実行確率をモデル化である．TF 機構が実行される割合を予測するアルゴリズムを構成し，ゼロワン過剰ベータ分布 (Zero-One-inflated Beta distribution) にもとづくモデル化を行った．各ノードの TF 機構の実行割合は，時刻の範囲外の影響により予測精度が悪化し，小さめに予測される傾向にある．そのため，新たなパラメータ k_U を導入し，k_U を超える出次数となるノードで $\beta = 1$ と補正する．$\beta = 1$ は 1 つ目の引用を PA で 2 回目以降は全て TF を実行することになるが，これは先行研究 [Krapivsky and Redner, 2005] が指摘された “文献引用のコピー” と類似の構造と捉えることができる．特許文献の引用ネットワークと学術論文の引用ネットワークに対して，シミュレーションを用いた実験を行った結果，拡張 (1) と (2) を組みあわせた YN+Ex(1)+(2) モデルが，どちらの引用ネットワークに対しても最も高い適合を確認できた．

本章では特許文献の引用ネットワークのうち，着目したサブカテゴリのみの特徴を捉えた生成モデルを取り扱った．既存のモデルに対し 2 種類の拡張を適用することで，一般化を進めることができたと考えている．しかしながら対象の引用ネットワークは複数のカテゴリとサブカテゴリが階層的にクラスタ構造を構成しているため，全体をモデリングするためにはクラスタ構造同士の接続を考慮した生成機構が必要になる．カテゴリ間やサブカテゴリ間を横断する引用は相対的に小さいものの，これまでモデリングの対象ではない構造となるため，今後の課題といえる．

第 5 章

おわりに

本論文では著名な書誌データベース Web of Science (WoS) に含まれる学術論文，特許文献からの構築された引用ネットワークを対象に，生成モデルの構築に関する研究を行った．

第 1 章では，本研究における学術的な背景について述べ，関連のある先行研究について説明した．

第 2 章では，引用構造がどのようにネットワークで表現できるかを説明し，利用可能な引用ネットワークデータを紹介した．その後，ネットワーク構造に対するいくつかの特徴量を紹介した．引用ネットワークでは引用数や被引用数に対応する次数や，三角形型の引用構造の密集などが特に注目するべき特徴量である．また対象のネットワークに類似となる構造を生成可能な生成モデルに関する先行研究について紹介した．第 3 章や第 4 章で，比較に用いる成長モデルを，優先的選択，三角形形成などの特徴的なエッジ生成の機構についてまとめた．

第 3 章では，学術論文の引用ネットワークに対する確率生成モデルを提案した．まず Web of Science 書誌データから確率統計分野の学術論文を抽出し，引用ネットワーク `WoS-Stat` を構築した．`WoS-Stat` に関して文献数の推移や，時刻ごとの引用の比率，次数分布や三角形数の分布などの基本的な性質を示した．その後，引用する論文がデータの範囲外となることに起因する，古い文献は新しい文献に比べて出次数が小さくなる点に着目し，時刻調整済みの特徴量である引用の年齢分布と出次数を定義した．これらの特徴量は引用ネットワークにおいて，時刻に依存しづらい性質であることを実験的に示した．さらに時刻ごとの文献数の分布がロジスティクス関数に，また年齢分布が逆ガウス分布に，時刻調整済み出次数が指数分布にそれぞれ従うことを仮定した関数にもとづいて，優先的選択，三角形形成によるエッジ生成を組み合わせた生成モデルを構成した．本モデルは引用ネットワークの各ノードには離散時刻で表現された発表時刻が与えられているものとし，シミュレーションでデータそのものを必要とならないことを前提にしている．第 2 章で説明を行ったネットワーク特徴量を用いて，データとのあてはまりを検証した．

第 4 章では，米国特許文献の引用ネットワークを対象に，第 3 章で提案した確率生成モデルを拡張した．対象の特許文献の引用ネットワークはカテゴリとサブカテゴリによる階層的なクラスタ構造を有しているものの，ネットワーク全体，カテゴリ内，サブカテゴリ内での次数や三角形数の分布に関するネットワーク特徴量は大きく差がないことを確認することができた．つまり 2 ノード間の特徴や 3 ノード間の特徴は，ネットワーク全体で変化しにくいことを示している．またこのクラスタ構造は

グラフ・クラスタリングによりある程度は抽出できることも確認した．第 3 章の提案モデルで用いる 3 つの関数については一定のあてはまりを示したものの，シミュレーションで得られたネットワーク構造はデータと比べて，三角形数の分布に関する適合が十分でない．1 点目の理由としては TF 機構が選択するノードの候補が十分に確保されていないためであると判断し，ノードの候補の拡張を行った．2 点目の理由としては TF 機構の実行確率を定数パラメータで与えているためであると判断し，ゼロワン過剰ベータ分布にもとづくモデル化を行った．その際に必要となる，与えられたグラフが PA 機構と TF 機構で生成されたと仮定したときに，TF 機構の実行確率を予測するアルゴリズムを構成して，実行確率の性質を観察した．構成したアルゴリズムが実行確率を小さめに予測することを踏まえて，出次数がある値以上になったときに，実行確率を 1 に固定する補正を加えた．拡張された生成モデルは拡張前のモデルを含み，より柔軟な表現が可能となり，米国特許文献の引用ネットワークや学術論文の引用ネットワークに対して適合を改善した．

本研究で十分に検討できなった点についてまとめる．第 3 章と第 4 章での提案によりすでに離散時刻の明示的な考慮，シミュレーションにおけるデータ使用の排除，実用的な生成モデルを構築することができた．これらは既存モデルの課題を解決したものと捉えている．しかしながら米国特許文献の引用ネットワークにおいては，あるクラスタ構造となるサブカテゴリの一部を表現できただけに過ぎない．また書誌データベース Web of Science に含まれる学術論文においても，複数の分野を対象にした際にそのような構造は現れる可能性は高いと考えている．そのため引用ネットワーク全体の構造を表現するためには，複数のクラスタ構造を接続する機構が必要となるものの，本研究では検討ができていない．また現時点では提案モデルはを用いたシミュレーションは `WoS-Stat` で 10 分以上，`cit-Patents` 全体では数時間以上の実行時間が必要となる．このことからシミュレーションの高速化は重要な観点の 1 つといえる．加えて現状のシミュレーション・アルゴリズムは比較的，並列化に不向きであることから，精度への影響を抑えつつ並列化を意識したアルゴリズムへ再設計する必要があると考える．すでに PA 機構に関する高速化 [Batagelj and Brandes, 2005, Sanders and Schulz, 2016] はすでに存在するものの，TF 機構や本論文で扱ったモデルへの適用は十分に議論されていないため，今後の課題といえる．

本研究から将来的に派生する可能性について例をいくつか挙げる．まず 2 章で取りあげたリンク予測への適用である．リンク予測は生成モデルと比べて問題設定に差異があるものの，エッジ生成の機構は関連が強く，本研究の知見が適用できる可能性がある．つづいて文献への評価指標への適用である．観測された現時点の評価だけでなく，モデルによる今後生じるネットワーク構造の予測を組み合わせることで，文献への評価指標の表現力の向上へ寄与できる可能性がある．予測結果と実際の成長の類似性や相違性などを比較・整理することで，分析の示唆を得られることが期待される．

本研究では引用ネットワークという観点で観測された，知識構造の成長や発展のモデリングを対象としている．本研究の取り組みも，文献とその引用構造で表現した引用ネットワーク，著者とその共著関係で表現した共著ネットワーク，さらに文献とその内容に関する紐付けなど様々なネットワーク構造を個別にモデリングすることに対応している．すでにそれらのネットワークは知識グラフ（Knowledge graph）として構造化が一般的になりつつあり，グラフ畳み込みネットワーク (Graph Convolutional Network) やグラフニューラルネットワーク (Graph Neural Network) を用いて，有

用な特徴量を抽出することが可能になってきた [Hogan et al., 2021]．今後，統合された知識構造の成長や発展を総合的に扱ったモデリングに取り組む際に，本研究の成果が何らかの足がかりを担うことを期待したい．

参考文献

[Ammar et al., 2018] Ammar, W., Groeneveld, D., Bhagavatula, C., Beltagy, I., Crawford, M., Downey, D., Dunkelberger, J., Elgohary, A., Feldman, S., Ha, V., Kinney, R., Kohlmeier, S., Lo, K., Murray, T., Ooi, H.-H., Peters, M., Power, J., Skjonsberg, S., Wang, L., Wilhelm, C., Yuan, Z., van Zuylen, M., and Etzioni, O. (2018). Construction of the literature graph in semantic scholar. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 84–91, New Orleans - Louisiana. Association for Computational Linguistics.

[Barabási and Albert, 1999] Barabási, A.-L. and Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512.

[Batagelj and Brandes, 2005] Batagelj, V. and Brandes, U. (2005). Efficient generation of large random networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 71.

[Blondel et al., 2008] Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008.

[Bonacich, 1987] Bonacich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182.

[Brandes, 2001] Brandes, U. (2001). A faster algorithm for betweenness centrality. *The Journal of Mathematical Sociology*, 25(2):163–177.

[Brandes, 2008] Brandes, U. (2008). On variants of shortest-path betweenness centrality and their generic computation. *Social Networks*, 30(2):136–145.

[Brin and Page, 1998] Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30:107–117.

[Chang et al., 2019] Chang, L. L.-H., Phoa, F. K. H., and Nakano, J. (2019). A new metric for the analysis of the scientific article citation network. *IEEE Access*, 7:132027–132032.

[Chang et al., 2021] Chang, L. L.-H., Phoa, F. K. H., and Nakano, J. (2021). A generative model of article citation networks of a subject from a large-scale citation database. *Scientometrics*, 126:7373–7395.

[Clarivate Analytics, 1997] Clarivate Analytics (1997). Web of science.

[Cornell University, 1991] Cornell University (1991). arxiv.org.

[Erdös and Rényi, 1959] Erdös, P. and Rényi, A. (1959). On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290.

[Erdos and Renyi, 1960] Erdos, P. and Renyi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hungary. Acad. Sci.*, 5:17–61.

[Farkas et al., 2001] Farkas, I. J., Derényi, I., Barabási, A. L., and Vicsek, T. (2001). Spectra of "real-world" graphs: Beyond the semicircle law. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 64(2):12.

[Freeman, 1977] Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41.

[Garfield, 1955] Garfield, E. (1955). Citation indexes for science: A new dimension in documentation through association of ideas. *Science*, 122(3159):108–111.

[Girvan and Newman, 2002] Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826.

[Golosovsky and Solomon, 2017] Golosovsky, M. and Solomon, S. (2017). Growing complex network of citations of scientific papers: Modeling and measurements. *Physical Review E*, 95(1):1–26.

[Hagberg et al., 2008] Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. *7th Python in Science Conference (SciPy 2008)*, (SciPy):11–15.

[Hajra and Sen, 2005] Hajra, K. B. and Sen, P. (2005). Aging in citation networks. *Physica A: Statistical Mechanics and its Applications*, 346(1-2 SPEC. ISS.):44–48.

[Hirsch, 2005] Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences*, 102(46):16569–16572.

[Hogan et al., 2021] Hogan, A., Blomqvist, E., Cochez, M., D'Amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A. C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., and Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*, 54.

[Holme and Kim, 2002] Holme, P. and Kim, B. J. (2002). Growing scale-free networks with tunable clustering. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 65(2):2–5.

[Hosking and Wallis, 1987] Hosking, J. R. M. and Wallis, J. R. (1987). Parameter and Quantile Estimation for the Generalized Pareto Distribution. *Technometrics*, 29(3):339.

[Karrer and Newman, 2011] Karrer, B. and Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 83.

[Krapivsky and Redner, 2005] Krapivsky, P. L. and Redner, S. (2005). Network growth by copying. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 71(3):1–7.

[Leskovec et al., 2010] Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., and Ghahramani, Z. (2010). Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research*, 11:985–1042.

[Leskovec and Krevl, 2014] Leskovec, J. and Krevl, A. (2014). SNAP Datasets: Stanford large network dataset collection.

[Leskovec and Sosič, 2016] Leskovec, J. and Sosič, R. (2016). Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(1):1.

[Liben-Nowell and Kleinberg, 2003] Liben-Nowell, D. and Kleinberg, J. (2003). The link prediction problem for social networks. pages 556–559. ACM.

[Linyuan and Zhou, 2011] Linyuan, L. L. and Zhou, T. (2011). Link prediction in complex networks: A survey.

[Mahadevan et al., 2006] Mahadevan, P., Krioukov, D., Fall, K., and Vahdat, A. (2006). Systematic topology analysis and generation using degree correlations. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '06, page 135–146, New York, NY, USA. Association for Computing Machinery.

[Newman et al., 2001] Newman, M. E., Strogatz, S. H., and Watts, D. J. (2001). Random graphs with arbitrary degree distributions and their applications. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 64(2):17.

[Oliveira et al., 2018] Oliveira, R. I., Ribeiro, R., and Sanchis, R. (2018). Disparity of clustering coefficients in the holme-kim network model.

[Orsini et al., 2015] Orsini, C., Dankulov, M. M., Colomer-De-Simon, P., Jamakovic, A., Mahadevan, P., Vahdat, A., Bassler, K. E., Toroczkai, Z., Bogunã, M., Caldarelli, G., Fortunato, S., and Krioukov, D. (2015). Quantifying randomness in real networks. *Nature Communications*, 6(May).

[Ospina and Ferrari, 2010] Ospina, R. and Ferrari, S. L. (2010). Inflated beta distributions. *Statistical Papers*, 51:111–126.

[Price, 1976] Price, D. D. S. (1976). A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27(5):292–306.

[Redner, 2004] Redner, S. (2004). Citation Statistics From More Than a Century of Physical Review. pages 1–12.

[Sabidussi, 1966] Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, 31(4):581–603.

[Sanders and Schulz, 2016] Sanders, P. and Schulz, C. (2016). Scalable generation of scale-free

graphs. *Information Processing Letters*, 116:489–491.

[Seshadri, 1999] Seshadri, V. (1999). *The Inverse Gaussian Distribution*, volume 137 of *Lecture Notes in Statistics*. Springer New York, New York, NY.

[Sinha et al., 2015] Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-J. P., and Wang, K. (2015). An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15 Companion, page 243–246, New York, NY, USA. Association for Computing Machinery.

[Staudt et al., 2016] Staudt, C. L., Sazonovs, A., and Meyerhenke, H. (2016). Networkit: A tool suite for large-scale complex network analysis. *Network Science*, 4:508–530.

[Virtanen et al., 2020] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

[Watts and Strogatz, 1998] Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world'networks. *Nature*, 393(6684):440–442.

[Wernicke, 2006] Wernicke, S. (2006). Efficient detection of network motifs. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(4):347–359.

[Wu and Holme, 2009] Wu, Z. X. and Holme, P. (2009). Modeling scientific-citation patterns and other triangle-rich acyclic networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 80(3).

[Yasui et al., 2013] Yasui, Y., Fujisawa, K., and Goto, K. (2013). Numa-optimized parallel breadth-first search on multicore single-node system. *Proceedings - 2013 IEEE International Conference on Big Data, Big Data 2013*, pages 394–402.

[Yasui et al., 2011] Yasui, Y., Fujisawa, K., Goto, K., Kamiyama, N., and Takamatsu, M. (2011). Netal: High-performance implementation of network analysis library considering computer memory hierarchy. *Journal of the Operations Research Society of Japan*, 54(4):259–280.

[Yasui and Nakano, 2021] Yasui, Y. and Nakano, J. (2021). A model to express citation relationships among academic papers. In *NETWORKS 2021*.

[Yasui and Nakano, 2022a] Yasui, Y. and Nakano, J. (2022a). Data from: A stochastic generative model for citation networks among academic papers.

[Yasui and Nakano, 2022b] Yasui, Y. and Nakano, J. (2022b). A stochastic generative model for citation networks among academic papers. *PLOS ONE*, 17(6):1–16.

[Yee, 2015] Yee, T. W. (2015). *Vector Generalized Linear and Additive Models*. Springer New York.

[Zhang and Chen, 2018] Zhang, M. and Chen, Y. (2018). Link prediction based on graph neural networks. *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 5171–5181.

付録

A　関連モデルのシミュレーションのためのアルゴリズム

モデルごとのシミュレーションに用いるアルゴリズムについて説明を行う．

A.1　初期化

まず Algorithm 4 は以降のアルゴリズムで共通した初期状態を構築する．指定した次数 k に対応しノード数 k, エッジ数 $k+1$ からなる連結構造を構築することができる．

Algorithm 4: 初期状態の構築

Input: ノード数 n, 出次数 k

Result: (V, E)

Procedure `Initialize`(k)**:**

  $V \leftarrow \{v_1, ..., v_k\}$

  $E \leftarrow \{(v_k, v_1), ..., (v_k, v_{k-1})\}$

  return

A.2　Barabási–Albert モデル

Algorithm 5 は Barabási–Albert モデルを表したものである．優先的選択はすでにネットワーク上に存在するノード集合 V からノード v_j を $d_{\mathrm{in}}(v_j)+1$ に比例する確率で選択する．これを各ノードごとに指定された出次数 k 回繰り返す．

A.3　Holme–Kim モデル

Algorithm 6 は Holme–Kim モデルでシミュレーションを行うためのアルゴリズムである．初期化には Algorithm 4 を，優先的選択には Barabási–Albert モデルと共通の `PreferentialAttachment` を用いる．加えて三角形形成を実行する割合はパラメータ β で指定し，0 以上 1 未満の疑似乱数を生成する関数 `Random`$(0, 1)$ を用いて確率的に決定する．

Algorithm 5: Barabási–Albert モデル

Input: ノード数 n, 出次数 k

Result: (V, E)

Procedure `PreferentialAttachment`(V, v_i)**:**

  ノード $v_j \in V \setminus \{v_i\}$ を $d_{\mathrm{in}}(v_j) + 1$ に比例する確率で選択

  return v_j

Procedure `GenerateBAnetwork`(n, k)**:**

  (V, E) を `Initialize`(k) で初期化

  for $i \leftarrow k + 1$ **to** n **do**

    $V \leftarrow V \cup \{v_i\}$

    for 1 **to** k **do**

      $v_j \leftarrow$ `PreferentialAttachment`(V, v_i)

      $E \leftarrow E \cup \{(v_i, v_j)\}$

  return (V, E)

Algorithm 6: Holme–Kim モデル

Input: ノード数 n, 出次数 k, TF を選択する割合パラメータ β

Result: (V, E)

Procedure `TriadFormation`(v_i, v_j)**:**

  ノード $v_k \in A(v_j) \setminus \{v_i\}$ をランダムに選択

  return v_k

Procedure `GenerateHKnetwork`(n, k, β)**:**

  (V, E) を `Initialize`(k) で初期化

  for $v_i \leftarrow k + 1$ **to** n **do**

    $V \leftarrow V \cup \{v\}$

    $v_j \leftarrow \emptyset$

    for 1 **to** k **do**

      if $v_j \neq \emptyset$ **or** `Random`(0,1) $< \beta$ **then**

        $v_k \leftarrow$ `TriadFormation`(v_i, v_j)

        $E \leftarrow E \cup \{(v_i, v_k)\}$

        contitnue

      $v_j \leftarrow$ `PreferentialAttachment`(V, v_i)

      $E \leftarrow E \cup \{(v_i, v_j)\}$

  return (V, E)

A.4 Wu–Holme モデル

Algorithm 7 は Wu–Holme モデルでシミュレーションを行う際に用いるアルゴリズムである．Wu–Holme モデルの `PreferentialAttachment` は Barabási–Albert や Holme–Kim と異なり，指数関数に従う経時効果（Aging effect）に従う確率でノードを選択する．一方 `TriadFormation` は Wu–Holme と同じものを用いる．

Algorithm 7: Wu–Holme モデル

Input: ノード数 n, 出次数 $\{k_i\}$, $i \in \{1, 2, ..., n\}$, 経年変化を表すパラメータ α, TF 機構を選択する確率 β

Result: (V, E)

Procedure `PreferentialAttachment`(V, v_i, α):
    ノード $v_j \in V$ を α^{i-j} に比例する確率で選択
    return v_j
Procedure `GenerateWHnetwork`(n, $\{k_i\}$, α, β):
    (V, E) を `Initialize`(k_1) で初期化
    for $i \leftarrow 1$ **to** n **do**
        $V \leftarrow V \cup \{v\}$
        $v_j \leftarrow \emptyset$
        for 1 **to** k **do**
            if $v_j \neq \emptyset$ **or** `Random`(0,1) $< \beta$ **then**
                $v_k \leftarrow$ `TriadFormation`(f_c, v_i, v_j)
                $E \leftarrow E \cup \{(v_i, v_k)\}$
                contitnue
            $v_j \leftarrow$ `PreferentialAttachment`(V, v_i, α)
            $E \leftarrow E \cup \{(v_i, v_j)\}$
    return (V, E)

B　本研究で用いた引用ネットワークの利用方法

B.1　WoS-Stat の利用方法

本研究で用いた WoS-Stat は Dryad データレポジトリ [Yasui and Nakano, 2022a] にて公開されている．データは wos-stat_nodes.csv と wos-stat_edges.csv からなり，Listing 5.1, 5.2 に先頭の 5 行のみまとめる．各ノード ID に対応する node_id 列は publication_year 列と uid 列によりソートしたノードに対して，0, 1, ..., と付与したものである．

Listing 5.1. wos-stat_nodes.csv

```
node_id,publication_year,uid
0,1981,WOS:A1981KW30100002
1,1981,WOS:A1981KW30100003
2,1981,WOS:A1981KW30100004
3,1981,WOS:A1981KW30100005
4,1981,WOS:A1981KW30100007
```

Listing 5.2. wos-stat_edges.csv

```
from,to
14,15
34,33
41,95
141,140
267,268
```

また Listing 5.3 は Python 言語で該当データを読み込むための最小コードである．読み込みには Pandas の DataFrame 構造を用いて，ネットワーク分析のためのライブラリ NetworkX に格納する．

Listing 5.3. wos-stat_edges.csv

```
import pandas as pd
import networkx as nx
edges = pd.read_csv("wos-stat_edges.csv")
nodes = pd.read_csv("wos-stat_nodes.csv", index_col="node_id")
G = nx.from_pandas_edgelist(edges, source="from", target="to", create_using=nx.
  DiGraph)
nx.set_node_attributes(G, nodes.to_dict("index"))
```

B.2 arXiv-HepTh と arXiv-HepPh における文献 ID から時刻情報の抽出

arXiv の書誌データから構築した引用ネットワークデータ hep-th と hep-ph は [Leskovec and Krevl, 2014] にて公開されている．

hep-th は各レコードに「引用元文献 ID，引用先文献 ID」となる引用データのほか，「文献 ID，発表時刻」となる公開時刻データが公開されている．一方 hep-ph は引用データのみで，公開時刻データを持たない．しかしながらいずれも文献 ID は YYMMNNN という形式となり，このうち YY は発表年の下二桁に, MM は発表月と発表時刻情報に対応する．一部，さらに先頭のゼロの欠損が推測されたため YYMMNNN に従ってゼロを補完する．Listing 5.4 の Python スクリプトは論文の文献 ID から時刻情報を抽出するためのものである．

Listing 5.4. 論文 ID の補完

```
import pandas as pd

def correct_arxiv_atcl_id(x):
    x = str(x)[:]
    yy, mm, nnn = x[0:-5], x[-5:-3], x[-3:]
    if not yy:
        yy = "00"
    _id = "{:02d}{:02d}{:03d}".format( int(yy), int(mm), int(nnn) )
    return _id

# for cit-HepTh
df = pd.read_csv("cit-HepTh.txt.gz", sep="\t", comment="#", names=["vi_", "vj"])

# for cit-HepPh
df = pd.read_csv("cit-HepPh.txt.gz", sep="\t", comment="#", names=["vi_", "vj"])

df["vi"] = df["vi_"].apply(correct_arxiv_atcl_id)
df["vj"] = df["vj_"].apply(correct_arxiv_atcl_id)
df["pt_i"] = pd.to_datetime(df["vi"].str[:4], format="%y%m")
df["pt_j"] = pd.to_datetime(df["vj"].str[:4], format="%y%m")
df["nn_i"] = df["vi"].str[4:]
df["nn_j"] = df["vj"].str[4:]

# >>> df.head(10)
#       vi_       vj_        vi        vj       pt_i       pt_j  nn_i  nn_j
# 0   1001   9304045   0001001   9304045 2000-01-01 1993-04-01   001   045
# 1   1001   9308122   0001001   9308122 2000-01-01 1993-08-01   001   122
# 2   1001   9309097   0001001   9309097 2000-01-01 1993-09-01   001   097
# 3   1001   9311042   0001001   9311042 2000-01-01 1993-11-01   001   042
# 4   1001   9401139   0001001   9401139 2000-01-01 1994-01-01   001   139
```

Listing 5.4 により，得られた Pandas DataFrame 構造を NetworkX の有向グラフ構造へ変換するための Python スクリプトを Listing 5.5 にまとめる．

Listing 5.5. NetworkX のグラフ構造へ入力

```python
import networkx as nx
from dateutil.relativedelta import relativedelta
def relative_quarter(d_t, d_0):
    t = relativedelta(d_t, d_0)
    return ( t.years * 12 + t.months ) // 3

# for cit-HepTh
lb, ub = pd.to_datetime("1992/01/01"), pd.to_datetime("2002/12/01")

# for cit-HepPh
lb, ub = pd.to_datetime("1993/01/01"), pd.to_datetime("2002/12/01")

X = df[
    ( lb <= df["pt_i"] ) & ( df["pt_i"] <= ub ) &
    ( lb <= df["pt_j"] ) & ( df["pt_j"] <= ub )
]
nodes = pd.concat([
  X[["vi","pt_i","nn_i"]],
  X[["vj","pt_j","nn_j"]].rename(
      columns={"vj": "vi", "pt_j": "pt_i", "nn_j": "nn_i"}
  )
]).drop_duplicates()
nodes = node_ids.sort_values(["pt_i", "nn_i"]).reset_index(drop=True)
nodes["vi_nm"] = node_ids.index

# >>> nodes.head(5)
#          vi          pt_i    nn_i    vi_nm
# 0    9201001    1992-01-01     001        0
# 1    9201002    1992-01-01     002        1
# 2    9201003    1992-01-01     003        2
# 3    9201004    1992-01-01     004        3
# 4    9201005    1992-01-01     005        4

pt_min = nodes["pt_i"].min()
nodes["tq_i"] = nodes["pt_i"].apply(lambda x: relative_quarter(x, pt_min))
nodes["atcl_i"] = nodes["vi"]
nodes = nodes.set_index("vi")

G = nx.from_pandas_edgelist(X, source="vi", target="vj", create_using=nx.DiGraph)
nx.set_node_attributes(G, nodes.to_dict("index"))
G = nx.relabel_nodes(G, { vi: vi_nm for vi, vi_nm in G.nodes(data="vi_nm") })
```

B.3 cit-Patents や cit-Patents-sc41 の利用方法

米国特許の引用ネットワークは NBER U.S. Patent Citations Data [*1] として公開されている．cit-Patents は引用データ acite75_99.zip と文献メタデータ apat63_99.zip を用いて，Listing 5.6 の処理で構築できる．

Listing 5.6. cit-Patents 引用ネットワークの構築

```
metas = pd.read_csv("apat63_99.txt", dtype="object")
metas = metas.fillna("")
# >>> metas[["PATENT", "GYEAR", "CAT", "SUBCAT"]].head(5)
#       PATENT   GYEAR   CAT   SUBCAT
# 0   3070801     1963     6        69
# 1   3070802     1963     6        63
# 2   3070803     1963     6        63
# 3   3070804     1963     6        63
# 4   3070805     1963     6        63

cites = pd.read_csv("cite75_99.txt", dtype="object")
# >>> cites.head(5)
#       CITING     CITED
# 0   3858241    956203
# 1   3858241   1324234
# 2   3858241   3398406
# 3   3858241   3557384
# 4   3858241   3634889

G_all = nx.DiGraph()
G_all.add_nodes_from([
    (m["PATENT"], dict(m))
    for m in metas.to_dict("records")
])
G_all.add_edges_from([
    (d["CITING"], d["CITED"])
    for d in cites.to_dict("records")
])
G_all.remove_nodes_from([
    vi
    for vi, ai in G_all.nodes(data=True)
    if "GYEAR" not in ai or "PATENT" not in ai
])
```

Listing 5.7 の処理により，指定したカテゴリ，指定したサブカテゴリの中にある引用ネットワーク構造を抽出する．cit-Patents-sc41 は以下の手順で抽出されたものである．

Listing 5.7. cit-Patents からカテゴリ（サブカテゴリ）内の引用ネットワークの抽出

```
def cat_subcat_graph(G_all, cat_i, cat_col="subcat"):
```

[*1] https://www.nber.org/research/data/us-patents

```
    G = nx.subgraph(
        G_all,
        [ vi for vi, ci in G_all.nodes(data=cat_col) if ci == cat_i ]
    )
    H = nx.DiGraph()
    H.add_nodes_from([
        (vi, ai)
        for vi, ai in G.nodes(data=True)
    ])
    H.add_edges_from([ (vi, vj) for vi, vj in G.edges() ])
    return H

G_cat4 = cat_subcat_graph(G_all, "4", cat_col="CAT")
G_subcat41 = cat_subcat_graph(G_all, "41", cat_col="SUBCAT")
```

B.4　提案モデルのシミュレーションのための参照実装

提案モデル [Yasui and Nakano, 2022b] を実現するための参照実装を Listing 5.8 と 5.9 にまとめる．まず Listing 5.8 の `generate` 関数でネットワーク生成を行う．`generate` 関数はシミュレーションのアルゴリズムの概要を実装し，PA 機構や TF 機構に関するノード選択は Listing 5.9 の `TimeAwareNetworkGrowth` クラスとして実装した．

`generate` 関数の引数である `fn` と `fc` は時刻 t ごとのノード数 $n(t)$ からなる $(t, \hat{f}_n(t))$ と，時刻 s ごとの引用の割合 $n(t)$ からなる $(s, \hat{f}_c(s))$ を，それぞれ `pandas.Series` 構造で表現したものである．また引数 `fo` は $\hat{f}_o$ から生成したノード v_i の出次数 $d^T_{\mathrm{out}}(v_i)$ からなる $(v_i, d^T_{\mathrm{out}}(v_i))$ を `numpy.array` 構造の `fo[i]` に対応させて表現したものである．`fb` は $\hat{f}_o$ から生成したノード v_i の TF 機構の実行確率 $p_{\mathrm{TF}}(v_i)$ からなる $(v_i, p_{\mathrm{TF}}(v_i))$ を `numpy.array` 構造の `fb[i]` に対応させて表現したものである．引数 `T_min` には開始時刻（時刻の最小値）を指定する．指定した時刻より前の時刻ではノード生成のみでエッジ生成を行わず，最後にこの範囲に含まれるノードやエッジは削除される．引数 `tf_cand_mode` は `"last"` を指定したときに TF 機構は直前の PA 機構で選択したノードの周辺のみを候補とする．それ以外では直前以外も候補とする．

Listing 5.8. 提案モデル [Yasui and Nakano, 2022b] のシミュレーションのための参照実装

```
import pandas as pd
import numpy as np
import random
import networkx as nx

def generate(fn, fc, fo, fb, time_attr="t", T_min=1, tf_cand_mode="all"):
    G = nx.DiGraph()
    gp = TimeAwareNetworkGrowth(fn, fc, tf_cand_mode=tf_cand_mode)

    for ti, n_ti in sorted( fn.items() ):
        print( ti, n_ti )
        U = np.arange(G.number_of_nodes(), G.number_of_nodes()+n_ti)
        G.add_nodes_from([ (vi, { time_attr: ti }) for vi in U ])
        gp.activate_nodes(ti, U)

        if ti < T_min:
            continue

        for vi, ki, beta in zip(U, fo[U], fb[U]):
            vj = -1
            for _ in range(ki):
                if not vj < 0 and random.random() < beta:
                    vk = gp.directed_triad_formation(G, vi)
                    if not vk < 0:
                        G.add_edge(vi, vk)
                        continue

                vj = gp.preferential_attachment(G, vi)
```

```
                if not vj < 0:
                    G.add_edge(vi, vj)

    G.remove_nodes_from([
        vi for vi, ti in G.nodes(data=time_attr) if ti < T_min
    ])
    return G
```

Listing 5.9. PA 機構と TF 機構からなるネットワーク成長

```
from random import choices
from collections import defaultdict

class TimeAwareNetworkGrowth(object):
    def __init__(self, fn, fc, T_min=0, tf_cand_mode=""):
        self.node_times = np.array([
          _t for _t, _n in fn.items() for _ in range(_n)
        ]).astype(int)
        self.fc = fc
        self.dt_max = self.fc.index.max()
        self.T_min = T_min
        self.weights = defaultdict(lambda: defaultdict(lambda: 0))
        self.use_last_pa = tf_cand_mode == "last"
        self.pa_nodes = set()

    def activate_nodes(self, ti, nodes):
        for vi in nodes:
            self.weights[ti][vi] += 1

    def weighted_random_choice(self, X):
        if not X:
            return -1
        Y = choices([ c for c, _ in X ], weights=[ w for _, w in X ], k=1)
        return Y[0]

    def random_choice_age(self, idx=[]):
        if len(idx) == 0:
            Y = choices(self.fc.index, weights=self.fc.values)
        else:
            Y = choices(self.fc.index[idx], weights=self.fc.values[idx], k=1)
        return Y[0]

    def preferential_attachment(self, G, vi):
        ti = self.node_times[vi]
        tj = ti - self.random_choice_age()
        if not self.weights[tj]:
            return -1
        succ_vi = set( G.successors(vi) )
        cands = [
            (_v, _c)
            for _v, _c in self.weights[tj].items()
            if _v not in succ_vi
        ]
```

```
        vj = self.weighted_random_choice(cands)
        if not vj < 0:
            if self.use_last_pa:
                self.pa_nodes = set()
            self.pa_nodes.add(vj)
            self.weights[tj][vj] += 1
        return vj

    def directed_triad_formation(self, G, vi, direction=(True,True)):
        ti = self.node_times[vi]
        cands = defaultdict(set)
        succ_vi = set(G.successors(vi)) | set([vi])
        for _vj in self.pa_nodes:
            for _bool, _iter in zip(
                direction, [ G.successors(_vj), G.predecessors(_vj) ]
            ):
                if _bool:
                    for vk in _iter:
                        if vk not in succ_vi:
                            cands[ self.node_times[vk] ].add( vk )
        dt_list = [
            ti-tk for tk in cands.keys() if 0 <= ti-tk <= self.dt_max
        ]
        vk = -1
        if dt_list:
            tk = ti - self.random_choice_age(idx=dt_list)
            vk_list = [ (_vk, self.weights[tk][_vk]) for _vk in cands[tk] ]
            vk = self.weighted_random_choice(vk_list)
        if not vk < 0:
            tk = self.node_times[vk]
            self.weights[tk][vk] += 1
        return vk
```