KH CoderとRを用いたネットワーク分析

著者	田中 京子
雑誌名	久留米大学コンピュータジャーナル
巻	28
ページ	37-52
発行年	2014-03-01
URL	http://hdl.handle.net/11316/529

［教 材 研 究］

KH CoderとRを用いたネットワーク分析
Network analysis with KH Coder and R

田中京子†
Kyoko Tanaka†

†久留米大学心理学研究科後期博士課程
† Graduate school of Psychology, Kurume University.

10. はじめに

心理学の研究において, 自由記述の質問紙調査やインタビューによってデータを取る場合がある。そのテキストデータを分析するために有効なものが計量テキスト分析である。計量テキスト分析では, 質的データにある種の数値化作業を加えることで計量的に分析する。そのために作られたソフトウェアに KH Coder*1がある[1]。

筆者は，人が人生において経験した出来事の記憶である自伝的記憶[2]の研究において，後期高齢者の自伝的記憶の特徴を探るために KH Coder を用いた。語られた思い出の中心となる語は共起ネットワークを用いて抽出し, ネットワークにおいてどれくらい中心的かを示す指標である中心性の数値の抽出は KH Coder に内蔵された R*2を利用した。

そこで，本稿では，KH Coder による共起ネットワーク分析と，R によるネットワークの中心性の数値の抽出について解説する。

11. 共起ネットワーク

テキストデータ内である語と他の語が一緒に出現することを共起といい, 共起する語を線で結んだものが共起ネットワークである。KH Coder の共起ネットワークでは，出現パターンの似通った語, すなわち共起の程度が強い語を線で結んだ共起ネットワークを描くことができる。

共起ネットワークは，中心性に基づいて，語の出現頻度や語と語の結びつきの程度に応じ，円の大きさや色あるいは円を結ぶ線の大きさによって表わされる。ただし，共起ネットワークでは，線で結ばれているかどうかが重要であり，近くに付置されているだけで，線で結ばれていなければ共起関係はない点に注意が必要である[3]。

2.1 分析対象ファイルの準備：テキストデータの HTML マーキング（図 2.1.1）

テキストデータのどこからどこまでを１つの文とみなすかを指定するため， HTML マーキ

*1 KH Coder は，計量テキスト分析もしくはテキストマイニングのためのフリーソフトウェアである。KH Coder の著作権は，樋口耕一が保持している。
*2 R は，数学や統計学のためのさまざまな関数やグラフィックス機能が備わっているデータ分析用のフリーソフトである。本稿で用いた R のバージョンは，R version 2.12.2(2011-02-25)である。

グを行い，メモ帳（notepad）に保存する。<h1></h1>は，後期高齢者の個人ごとの 10 代の自伝的記憶内容に対応している。

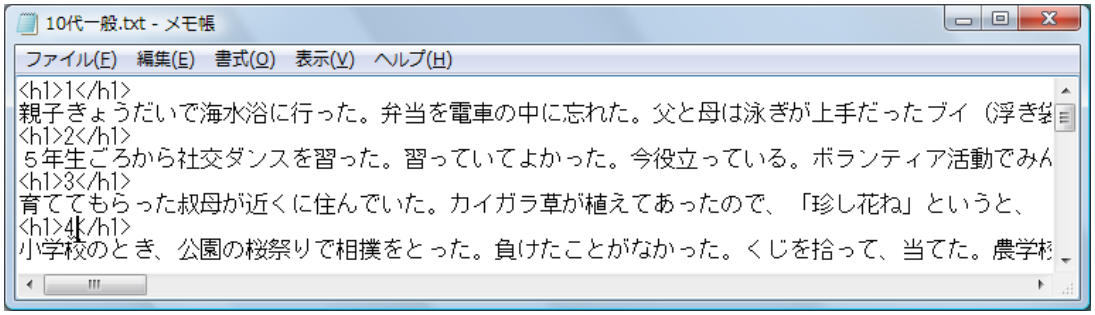


図 2.1　分析対象ファイル（一部）

2.2　分析対象ファイルの登録

分析対象ファイルを「プロジェクト」として KH Coder に登録する。

① KH Coder のメニューにある「プロジェクト」から「新規」をクリックし，「新規プロジェクト」の画面を開く。

② 「参照」をクリックし，分析対象ファイルを呼び込む。

③ 「OK」をクリックし，「プロジェクト・マネージャー」の画面を開く。

④ 分析対象ファイル（本稿では，「10 代一般」）を指定する。

⑤ 「開く」をクリックする。

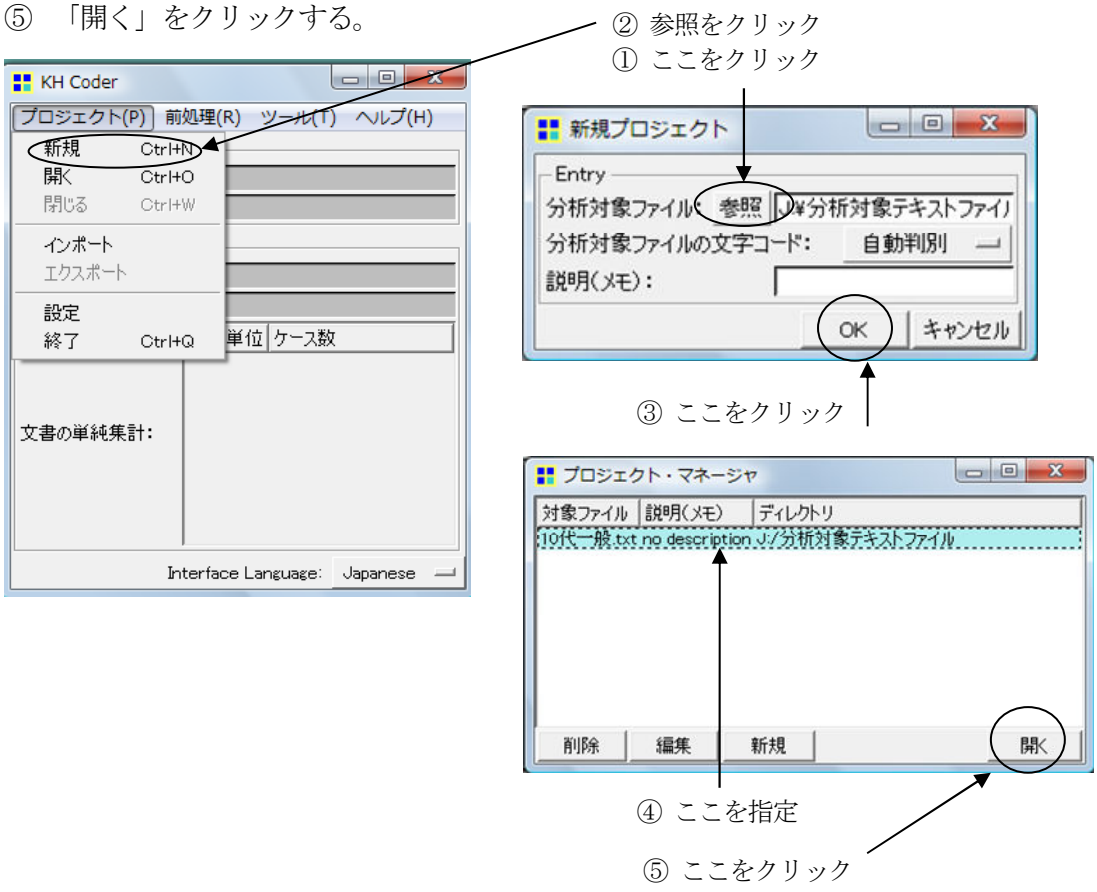


図 2.2.1　分析対象ファイルの登録

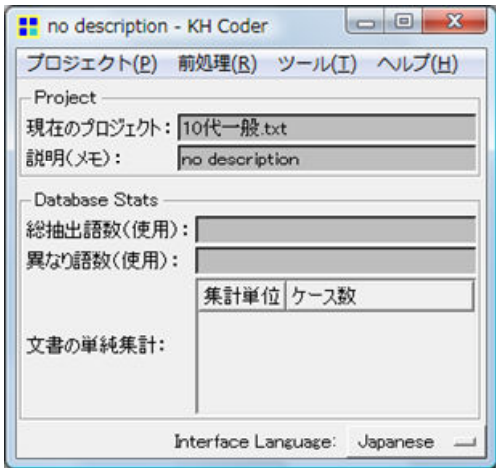


図 2.2.2　分析対象ファイル登録後の画面

なお，KH Coder のインストールと起動，形態素の出力等の基本的手順については，田中[4]を参照されたい。

2.3　共起ネットワークの起動（図 2.3）

① KH Coder のメニューバーにある「ツール」から「抽出語」を選択する。
② 「共起ネットワーク」をクリックする。

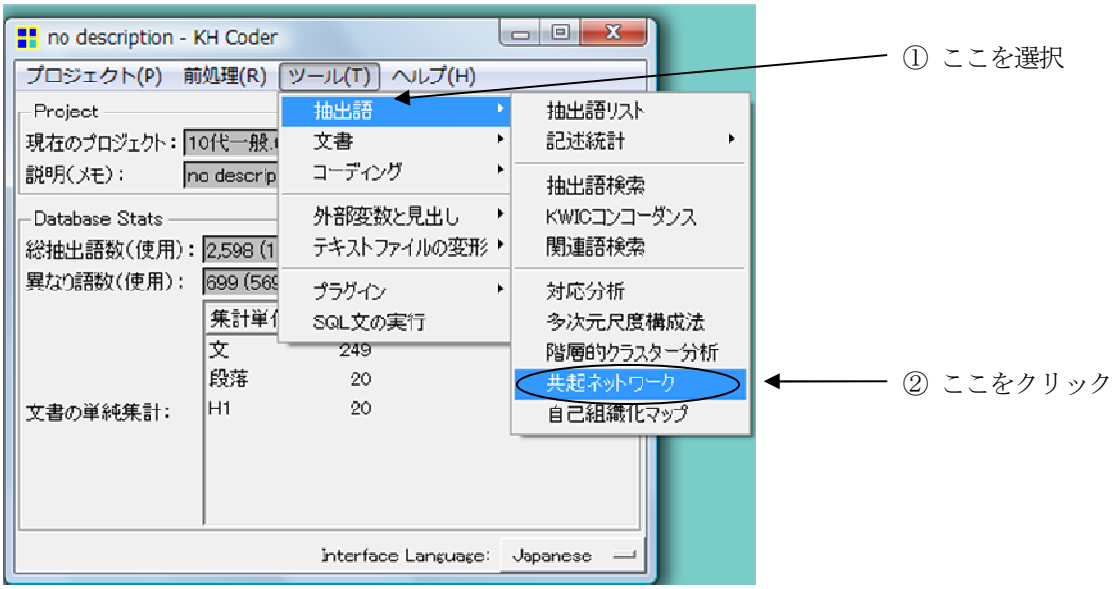


図 2.3　共起ネットワークの起動

2.4 共起ネットワークのオプション（図 2.4）

2.4.1 付置する語の選択と共起関係の絞り込み

すべての語をネットワーク図に付置するとネットワーク図が読み取れなくなる場合がある。そこで，出現数，文書数，品詞名で取捨選択を行い，付置する語の数を減らす必要がある。取捨選択は，分析の目的に応じて適宜編集すると良い。品詞名の☑を外すと，分析対象から外すことができる。林[5]は，品詞情報のうち「形容詞」,「名詞－サ変（サ変名詞)」,「名詞－一般」,「名詞－形容動詞語幹（形容動詞)」および「名詞－固有名詞－組織」を選択して分析している。取捨選択後の分析対象語数は,「チェック」ボタンをクリックすることで随時確認することができる。

筆者の研究では，最小出現数を 3 とし,「名詞」,「サ変名詞」,「形容動詞」,「形容詞」および「名詞 C（1 語の名詞)」を選択した。その結果，抽出された形態素（語）の数は 27 になった。

2.4.2 共起ネットワークの設定

共起関係（edge）の種類は，「語－語」と「語－外部変数・見出し」がある。語と語の結びつきを描く場合は「語－語」を選択し，見出しと頻出語とが互いにどのように結びついているかを描く場合は「語－外部変数・見出し」を選択する。本節では, 語と語がどのように結びついているのかを探るため，「語－語」を説明する。

共起関係（edge）をすべて線として描くと画面が線で埋まってしまうことが多い。そうした場合は，描画する共起関係を一部の比較的強いものに絞らなければならない。「描画する共起関係（edge）の絞り込み」の部分で「描画数」を選ぶと，Jaccard 係数[*3]の大きい順に指定された数の共起関係が選択・描画される。あるいは「Jaccard 係数」を選択すると，Jaccard 係数が指定された数よりも大きい共起関係がすべて選択・描画される。

① 最小出現数を入力する。
② 不要な品詞名のチェック（☑）を外す。
③ 「チェック」をクリックする。現在の設定で利用される語の数が表示される。
④ 「共起関係（edge）の種類」は,「語－語」を選択する。
⑤ 「描画する共起関係（edge）の絞り込み」は,「描画数」あるいは「Jaccard 係数」を選択する。
⑥ 「強い共起関係ほど太い線で描画」の場合は，✔を入れる。共起の程度に応じて共起関係を表す線（edge）の太さが変化する。
⑦ 「出現数の多いほど大きい円で描画」の場合は，✔を入れる。語の出現数に応じて，円のサイズが変化する。なお,「フォントも大きく」に✔を入れると，語の出現数に応じてフォントサイズも変化する。EMF・EPS・PDF 形式で保存し印刷する場合に使用するとよい。

[*3] 共起関係の強弱は，分析対象となった語のすべての組み合わせについて，「Jaccard の類似性測度（Jaccard 係数)」の値による。「0.1（軽い程度）～0.3（強い関連）程度」と考えられる[6]。

⑧ 「OK」をクリックする。

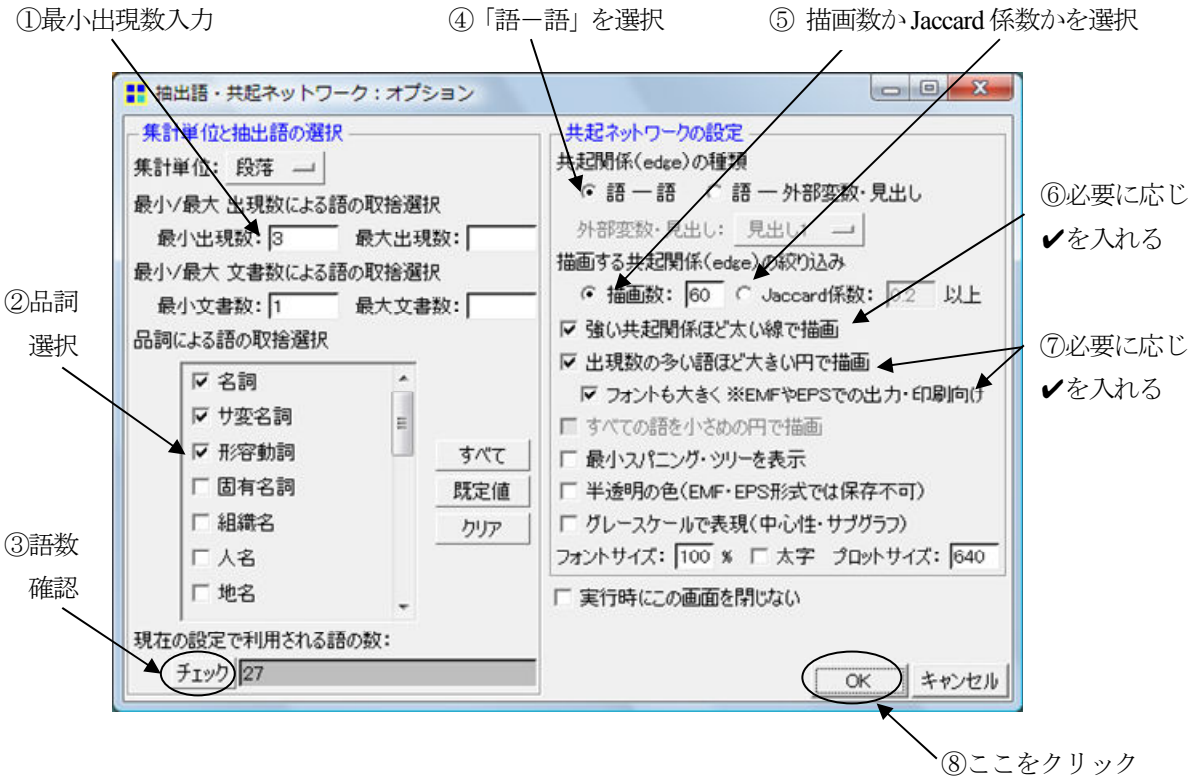


図 2.4　共起ネットワークのオプション

2.5　共起ネットワーク図の選択（図 2.5）

KH Coder では，語と語のネットワーク図が 6 種類用意されており，それらの中から選択できる。最初の 3 つは，「中心性（媒介）」，「中心性（次数）」，「中心性（固有ベクトル）」である。中心性に基づいて色分けが行われており，それぞれの語がネットワーク構造の中でどの程度中心的な役割を果たしているかを表す。水色・白・ピンクの順に中心性が高くなることを示す。

次の 3 つは，比較的強くお互いに結びついている部分を自動的に検出してグループ分けを行い，その結果を色分けによって示す「サブグラフ検出」[*4]である。共起関係の媒介性，random walks および modularity に基づいた方法の中から選ぶことができる。これらの色分けにおいては，背景が白で，丸い囲み枠が黒い色であれば，ほかの語とグループを形成していない単独の語であることを意味している。なお，同じサブグラフに含まれる語は実線で結ばれるのに対して，互いに異なるサブグラフに含まれる語は破線で結ばれる。

① 共起ネットワーク初期画面（図 2.5）左下の「中心性（媒介）」をクリックする。
② 「中心性（媒介）」，「中心性（次数）」，「中心性（固有ベクトル）」，「サブグラフ検出（媒介）」，

*4 「比較的強くお互いに結びついている部分」は，グラフ理論の分野で「コミュニティ」と呼ばれるが，KH Coder 上での表記は暫定的に「サブグラフ検出」とされている[3]。

「サブグラフ検出（random walks）」，「サブグラフ検出（modularity）」のうち，いずれかを選択する。

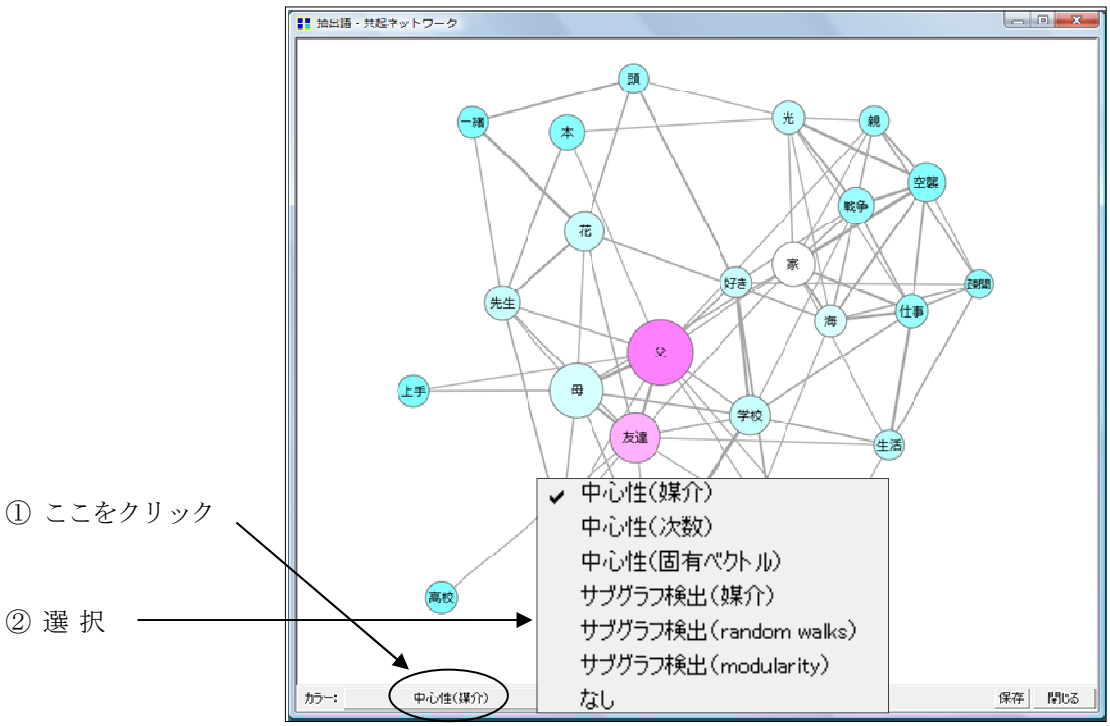


図 2.5　共起ネットワークの初期場面と選択場面

2.6　共起ネットワークにおける中心性

2.6.1　媒介中心性

媒介中心性は，さまざまなネットワークにおいて，他の頂点*5どうしの間にあって，それらをつなぐ働きをする頂点を見出そうとする中心性座標である[7]。つまり，他の頂点同士をつなぐ最短距離上に位置する頂点は，頂点間の仲介や情報のコントロールが可能な点で有力であり，より多くの頂点間の最短距離上にあるほど影響力が大きいと考える。

たとえば，頂点間の移動において中継地点になるという意味では交通の要衝，情報の迅速な伝達にかかわる点では情報通の個人をさす。

「父」や「友達」が語と語のつながりの中心であることが示されている。

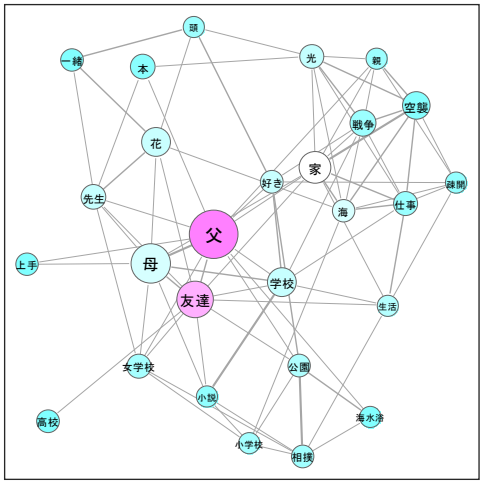


図 2.6.1　共起ネットワーク（媒介中心性）

*5 ネットワーク分析では，人間関係やウェブページのリンクなどの構造を，点（頂点あるいはノード）と線（辺あるいはリンク）によって構成される構造として抽象化してとらえる。社会ネットワークでは，たとえば，頂点は個人，辺は何らかの社会的関係を表し，ウェブページのネットワークであれば，個々のページが頂点，それらの間のリンクが辺として表される[7]。

2.6.2　次数中心性

次数中心性は，ある頂点に接続している辺の数，すなわち隣接する頂点の数に基づく中心性指標である[7]。

たとえば，友人関係ネットワークでは友人数が多い人，ウェブページのネットワークでは多くのリンクを集めているページの中心性が高く評価される。

他の語と結びつきが多い語は，「父」，「友達」，「家」，「母」，「学校」，「戦争」，「仕事」，「光」および「海」である。

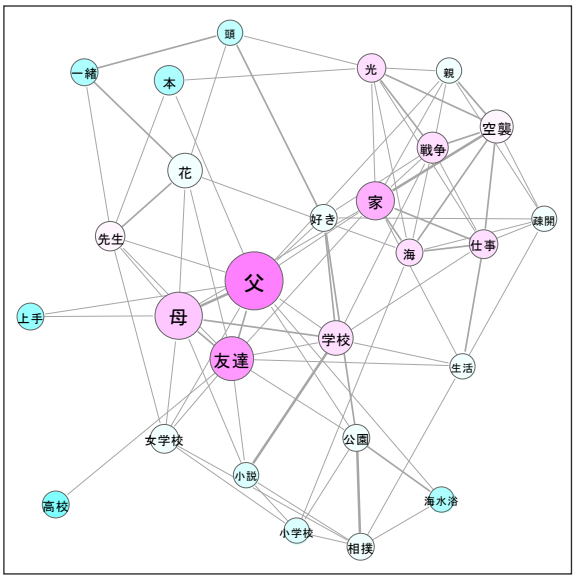


図 2.6.2　共起ネットワーク（次数中心性）

2.6.3　固有ベクトル中心性

固有ベクトル中心性では，ある頂点の中心性を評価するときに，頂点と隣接する頂点の中心性を反映させる[7]。

たとえば，友人の多い友人とのつながりは，そうでない人のつながりよりも意味を持つ。

他との結びつきが多い語につながっている語は，「家」，「空襲」，「仕事」，「戦争」，「海」，「父」，「学校」，「友達」，「母」および「光」である。

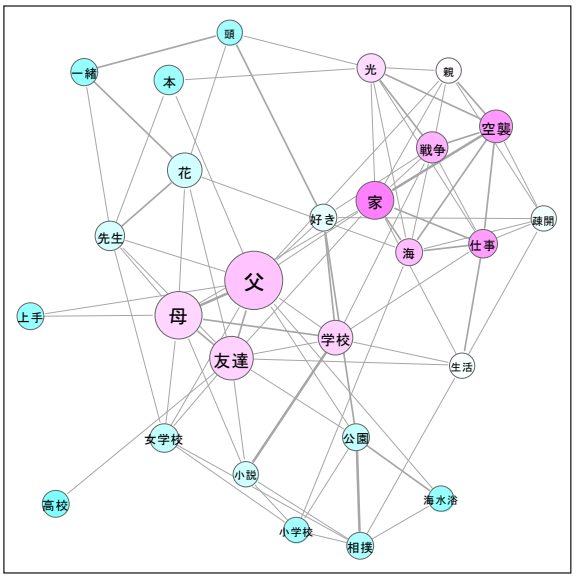


図 2.6.3　共起ネットワーク（固有ベクトル）

2.6.4 サブグラフ（媒介）

サブグラフ（媒介）は，辺の媒介中心性を用いたコミュニティにおけるサブグループ(サブグラフ)*6の抽出法である。点中心性の媒介中心性の算出方法を辺に適用したもので，ある点が頂点間の最短経路上にある程度を示し，媒介中心性の高い辺はそれだけ多くの頂点をつなぐ働きをしているといえる[7]。

媒介性に基づくサブグラフでは，4 個のグループが示された。なお，図中のグループごとに囲んだ楕円は筆者が追加したものである（以下，「random walks 」と「modularity」も同様)。

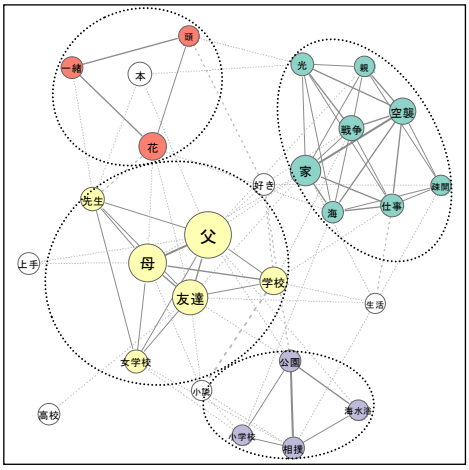


図 2.6.4 サブグラフ（媒介）

2.6.5 サブグラフ（random walks）

ランダム・ウォーク中心性は，媒介中心性を変更したもので，最短路の代わりに単純ランダム・ウォークを用いる[8]。伝達効率の悪いランダム・ウォークにより最短でない道の上にいる人にも中心性の部分点が与えられ[8]，たとえば，ランダム・ウォーク中心性では，最短距離よりも気まぐれな道筋で流れる情報を中継している人が中心的である[9]。

ランダム・ウォーク中心性によるサブグラフでは，5 つのグループが示された。

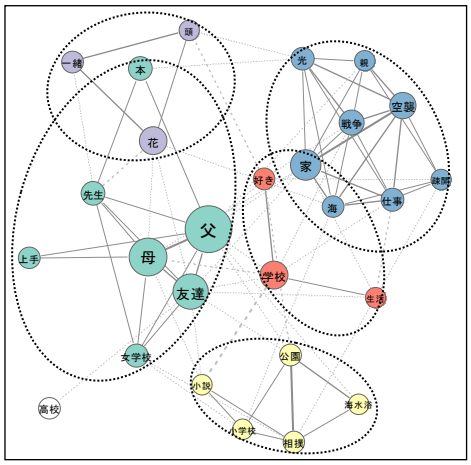


図 2.6.5 サブグラフ（random walks）

2.6.6 サブグラフ（modularity）

modularity（モジュラリティ）とは，分割されたコミュニティ内の辺の数とコミュニティ間の辺の数の比較により，コミュニティが高密度のサブグループをうまく抽出しているかを示す指標である[7]。

modularity によるサブグラフでは，4 個のグループがあることが示された。

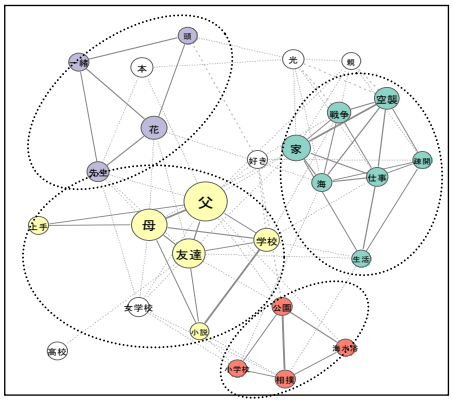


図 2.6.6 サブグラフ（modularity）

*6 前述したように，「比較的強くお互いに結びついている部分」であるコミュニティを，KH Coder 上での表記は暫定的に「サブグラフ検出」とされている[3]。また，ネットワーク内のサブグループはグラフ内部のサブグラフとして表現される[7]。そこで，本稿では，コミュニティの抽出法を用いて 3 種類のサブグラフを説明する。

3. 中心性の数値化

KH Coder では，共起ネットワークにおける中心性得点と，サブグラフのクラスター番号を確認することができる[6]。

3.1 「R Source」形式で保存（図 3.1）

① 共起ネットワーク図の左下の「保存」をクリックする。
② 「プロットを保存」の画面で「保存先」を指定する。（保存先：筆者の USB メモリー内の「中心性 R」フォルダー ）
③ 「ファイル名」を入力する。（ファイル名：10 代一般）
④ 「ファイルの種類」の中から「R Source(*.r)」を選択する。
⑤ 保存をクリックする。

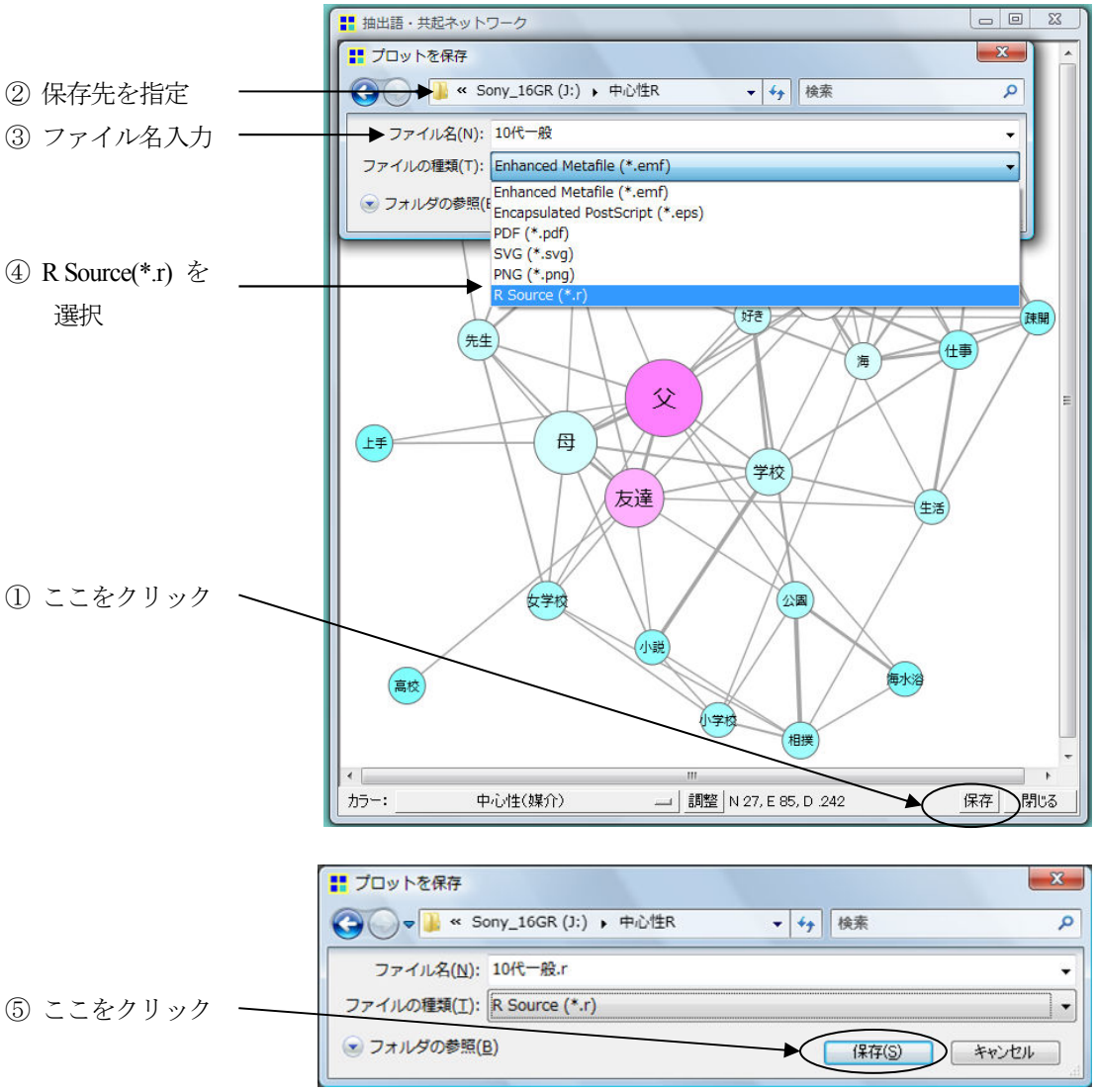


図 3.1 「R Source」形式で保存

3.2 R の実行

① KH Coder フォルダーの「Rgui.bat」をクリックすると，R の初期画面が表示される。

図 3.2.1　KH Coder のフォルダー画面

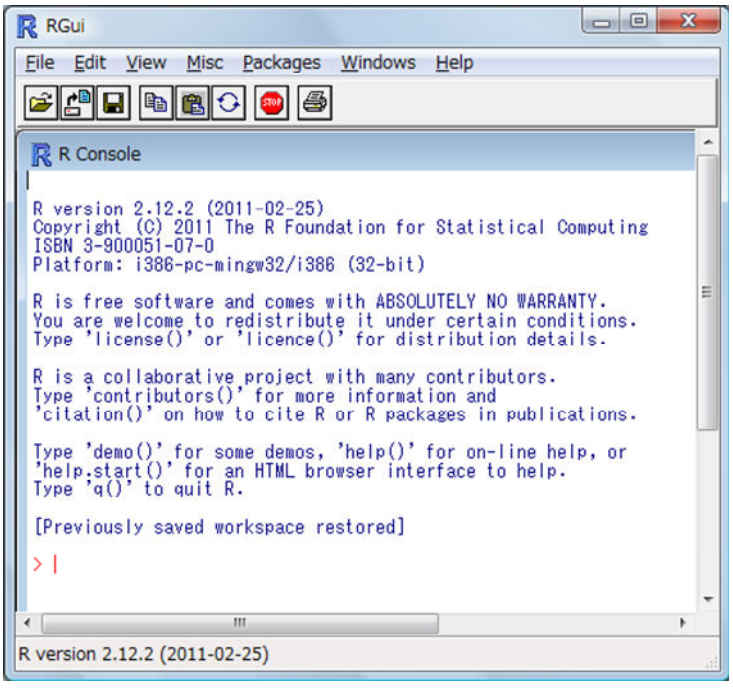


図 3.2.2　R の初期画面

② Rの初期画面の「File」から「Source R code」をクリックする。

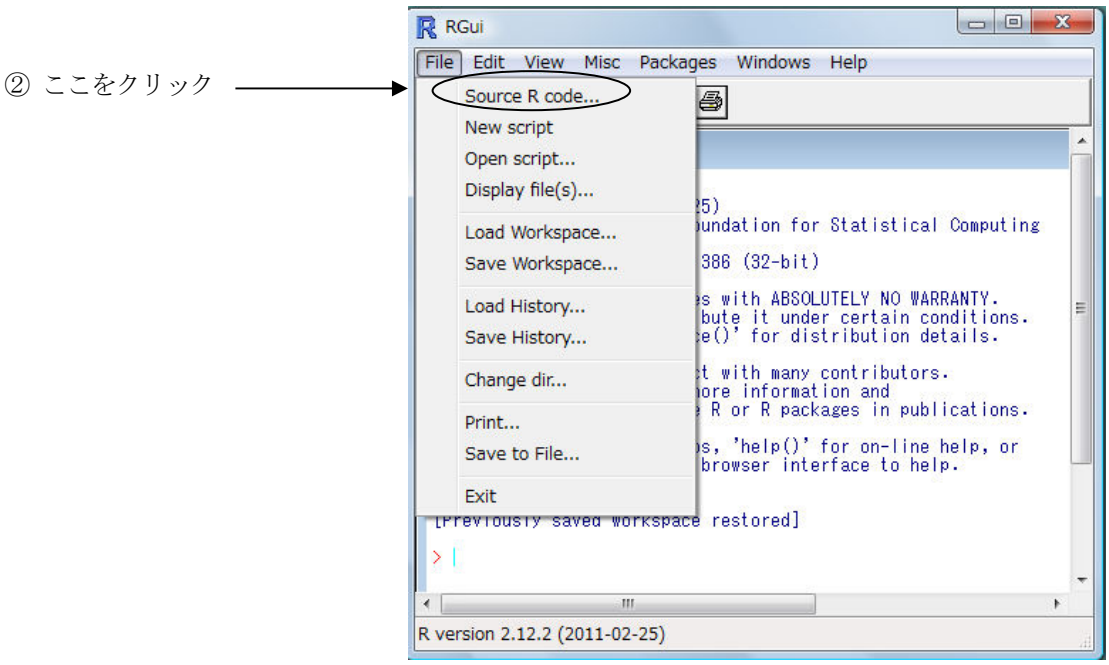


図 3.2.3　R コードの呼び込み

③ 「R Source」形式で保存したファイルを指定する。(本稿では,「中心性R」フォルダー内の「10代一般」)。
④ 「開く」をクリックする。

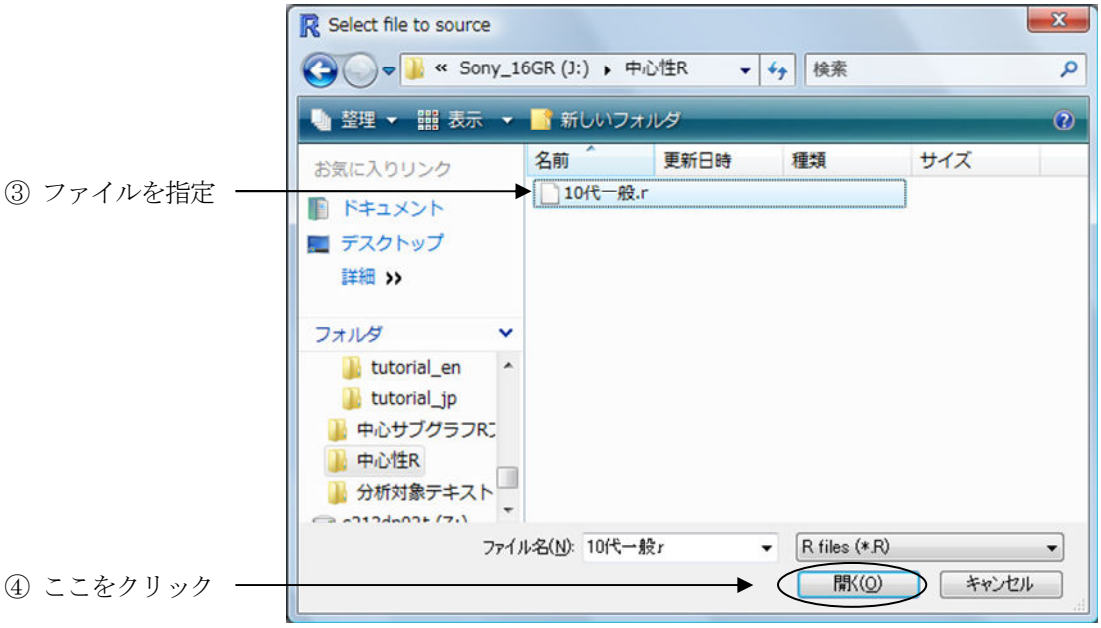


図 3.2.4　ファイルを開く

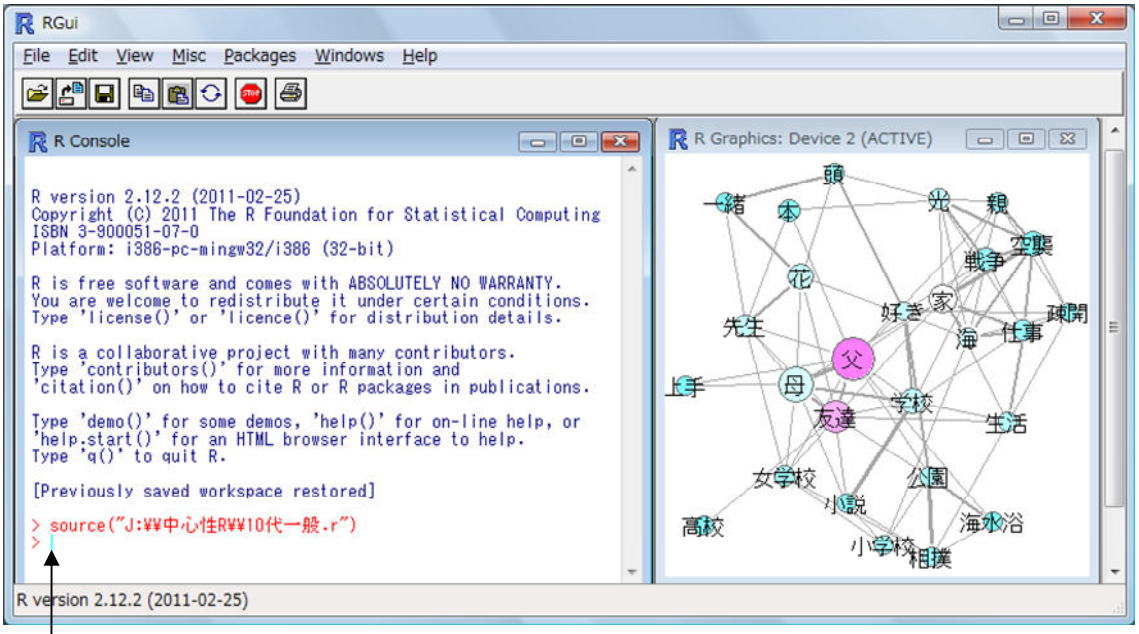


図 3.2.5　ファイルを開いた後の R の画面

⑤ ここに R コードを入力

⑤　R コード[6]（次ページの中心性得点とサブグラフのクラスター番号抽出用 R コード）を R 上に入力し，中心性の得点とサブグラフのクラスター番号を抽出する（図 3.2.7）。

なお，R コードは，R 上に直接入力してもよいが，メモ帳（notepad）にいったん保存した後，必要に応じてコピーし，R 上に貼り付けるほうが便利である（図 3.2.6）。

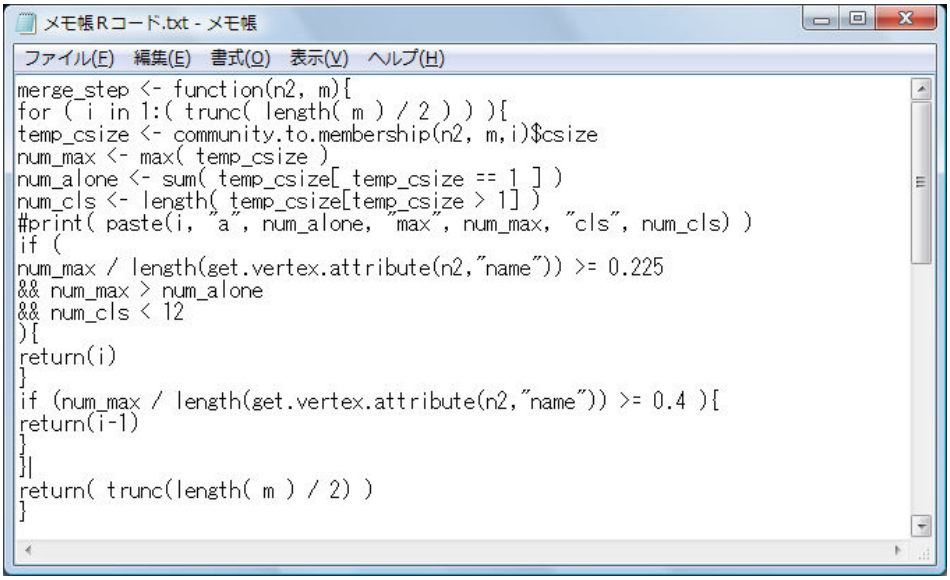
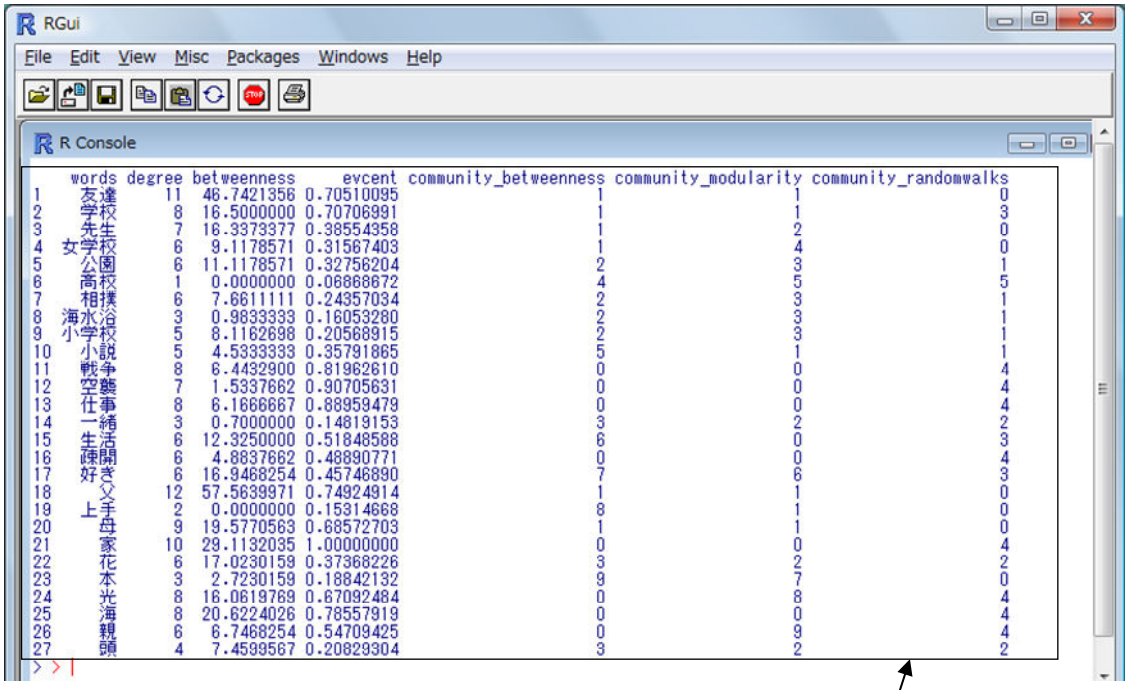


図 3.2.6　メモ帳に保存した R コードの一部

中心性得点とサブグラフのクラスター番号抽出用R コード

```
merge_step <- function(n2, m){
for ( i in 1:( trunc( length( m ) / 2 ) ) ){
temp_csize <- community.to.membership(n2, m,i)$csize
num_max <- max( temp_csize )
num_alone <- sum( temp_csize[ temp_csize == 1 ] )
num_cls <- length( temp_csize[temp_csize > 1] )
#print( paste(i, "a", num_alone, "max", num_max, "cls", num_cls) )
if (
num_max / length(get.vertex.attribute(n2,"name")) >= 0.225
&& num_max > num_alone
&& num_cls < 12
){
return(i)
}
if (num_max / length(get.vertex.attribute(n2,"name")) >= 0.4 ){
return(i-1)
}
}
return( trunc(length( m ) / 2) )
}

# コミュニティ検出(betweenness)
com_b <- edge.betweenness.community(n2, directed=F)
com_b <- community.to.membership(n2, com_b$merges, merge_step(n2,com_b$merges)
)

# コミュニティ検出(modularity)
com_m <- fastgreedy.community(n2, merges=TRUE, modularity=TRUE)
com_m <- community.to.membership(n2, com_m$merges, merge_step(n2,com_m$merges)
)

# コミュニティ検出(random walks)
com_r <- walktrap.community(n2,weights=get.edge.attribute(n2, "weight")
)

# 1つのデータフレームにまとめる
cnt <- data.frame(
words = colnames(d)[
as.numeric( get.vertex.attribute(n2,"name") )
],
degree = degree(n2),
betweenness = betweenness(n2),
evcent = evcent(n2)$vector,
community_betweenness = as.character(com_b$membership),
community_modularity = as.character(com_m$membership),
community_randomwalks = as.character(com_r$membership)
)

print(cnt)
```

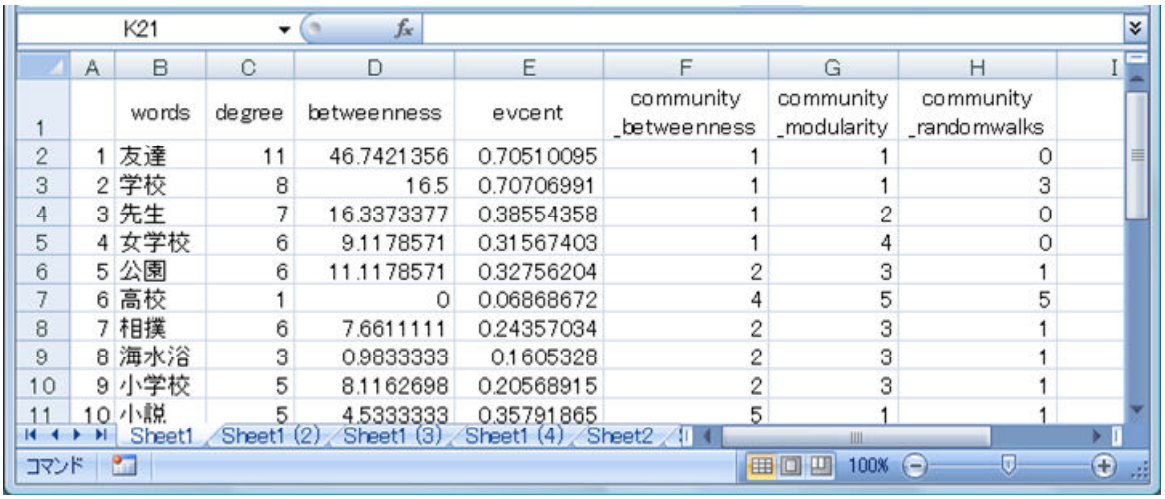


	words	degree	betweenness	evcent	community_betweenness	community_modularity	community_randomwalks
1	友達	11	46.7421356	0.70510095	1	1	0
2	学校	8	16.5000000	0.70706991	1	1	3
3	先生	7	16.3373377	0.38554358	1	2	0
4	女学校	6	9.1178571	0.31567403	1	4	0
5	公園	6	11.1178571	0.32756204	2	3	1
6	高校	1	0.0000000	0.06868672	4	5	5
7	相撲	6	7.6611111	0.24357034	2	3	1
8	海水浴	3	0.9833333	0.16053280	2	3	1
9	小学校	5	8.1162698	0.20568915	2	3	1
10	小説	5	4.5333333	0.35791865	5	1	1
11	戦争	8	6.4432900	0.81962610	0	0	4
12	空襲	7	1.5337662	0.90705631	0	0	4
13	仕事	8	6.1666667	0.88959479	0	0	4
14	一緒	3	0.7000000	0.14819153	3	2	2
15	生活	6	12.3250000	0.51848588	6	0	3
16	疎開	6	4.8837662	0.48890771	0	0	4
17	好き	6	16.9468254	0.45746890	7	6	3
18	父	12	57.5639971	0.74924914	1	1	0
19	上手	2	0.0000000	0.15314668	8	1	0
20	母	9	19.5770563	0.68572703	1	1	0
21	家	10	29.1132035	1.00000000	0	0	4
22	花	6	17.0230159	0.37368226	3	2	2
23	本	3	2.7230159	0.18842132	9	7	0
24	光	8	16.0619769	0.67092484	0	8	4
25	海	8	20.6224026	0.78557919	0	0	4
26	親	6	6.7468254	0.54709425	0	9	4
27	頭	4	7.4599567	0.20829304	3	2	2

図 3.2.7　抽出後の R の画面（一部）

⑥ R で検出された中心性得点とクラスター番号（実線で囲んだ部分）をコピーし，Excel 上にテキスト形式で張り付け，「テキストファイルウィザード」を使用してセルごとに張り付ける（図 3.2.8）。

「degree」は次数中心性に，「betweenness」は媒介中心性に，「evcent」は固有ベクトル中心性に対応している。さらに，「community_betweenness」はサブグラフ検出（媒介）に，「community_modularity 」はサブグラフ検出（modularity ）に，「community_randomwalk」はサブグラフ（randomwalks）に対応している。



	A	words	degree	betweenness	evcent	community_betweenness	community_modularity	community_randomwalks
2	1	友達	11	46.7421356	0.70510095	1	1	0
3	2	学校	8	16.5	0.70706991	1	1	3
4	3	先生	7	16.3373377	0.38554358	1	2	0
5	4	女学校	6	9.1178571	0.31567403	1	4	0
6	5	公園	6	11.1178571	0.32756204	2	3	1
7	6	高校	1	0	0.06868672	4	5	5
8	7	相撲	6	7.6611111	0.24357034	2	3	1
9	8	海水浴	3	0.9833333	0.1605328	2	3	1
10	9	小学校	5	8.1162698	0.20568915	2	3	1
11	10	小説	5	4.5333333	0.35791865	5	1	1

図 3.2.8　Excel に張り付けた中心性得点とクラスター番号（一部）

⑦ 中心性得点の高い順に並び変える。
「父」,「友達」,「光」,「母」,「家」,「学校」および「海」は，次数中心性，媒介中心性および固有ベクトル中心性ともに上位にあり，「戦争」と「空襲」は，次数中心性も固有ベクトル中心性も上位にある。
⑧ サブグラフの数値はクラスター番号を表しており，同じ番号は，同じグループに属する。番号ごとに並び変える。

表 3.2.9　中心性得点とサブグラフのクラスター番号

中心性得点							クラスター番号					
順位	words	次数中心性	words	媒介中心性	words	固有ベクトル中心性	words	サブグラフ媒介	words	サブグラフ modularity	words	サブグラフ randomwalks
1	父	12	父	57.564	家	1.000	戦争	0	戦争	0	友達	0
2	友達	11	友達	46.742	空襲	0.907	空襲	0	空襲	0	先生	0
3	家	10	家	29.113	仕事	0.890	仕事	0	仕事	0	女学校	0
4	母	9	海	20.622	戦争	0.820	疎開	0	生活	0	父	0
5	学校	8	母	19.577	海	0.786	家	0	疎開	0	上手	0
6	戦争	8	花	17.023	父	0.749	光	0	家	0	母	0
7	仕事	8	好き	16.947	学校	0.707	海	0	海	0	本	0
8	光	8	学校	16.500	友達	0.705	親	0	友達	1	公園	1
9	海	8	先生	16.337	母	0.686	友達	1	学校	1	相撲	1
10	先生	7	光	16.062	光	0.671	学校	1	小説	1	海水浴	1
11	空襲	7	生活	12.325	親	0.547	先生	1	父	1	小学校	1
12	女学校	6	公園	11.118	生活	0.518	女学校	1	上手	1	小説	1
13	公園	6	女学校	9.118	疎開	0.489	父	1	母	1	一緒	2
14	相撲	6	小学校	8.116	好き	0.457	母	1	先生	2	花	2
15	生活	6	相撲	7.661	先生	0.386	公園	2	一緒	2	頭	2
16	疎開	6	頭	7.460	花	0.374	相撲	2	花	2	学校	3
17	好き	6	親	6.747	小説	0.358	海水浴	2	頭	2	生活	3
18	花	6	戦争	6.443	公園	0.328	小学校	2	公園	3	好き	3
19	親	6	仕事	6.167	女学校	0.316	一緒	3	相撲	3	戦争	4
20	小学校	5	疎開	4.884	相撲	0.244	花	3	海水浴	3	空襲	4
21	小説	5	小説	4.533	頭	0.208	頭	3	小学校	3	仕事	4
22	頭	4	本	2.723	小学校	0.206	高校	4	女学校	4	疎開	4
23	海水浴	3	空襲	1.534	本	0.188	小説	5	高校	5	家	4
24	一緒	3	海水浴	0.983	海水浴	0.161	生活	6	好き	6	光	4
25	本	3	一緒	0.700	上手	0.153	好き	7	本	7	海	4
26	上手	2	高校	0.000	一緒	0.148	上手	8	光	8	親	4
27	高校	1	上手	0.000	高校	0.069	本	9	親	9	高校	5

4. おわりに

今回，後期高齢者の 10 代の自伝的記憶の特徴を探る方法の一つとして，KH Coder を用いて記憶内容のデータの抽出と解析を行った。語と語の関係性を共起ネットワーク図で示し，KH Coder に内蔵の R で中心性得点や語のまとまりであるクラスターを得た。

共起ネットワーク図を解釈する際，中心性の強さの程度を示す色の濃淡や，円の大きさの大小の判断が主観的になりがちになる。しかし，R で中心性得点を抽出することで，後高齢者の 10 代の自伝的記憶の特徴をより客観的に示すことができたといえる。

KH Coder には，語の共起関係を探索する方法として，ネットワーク分析の他に，対応分析やクラスター分析等の機能も備えつけられている。また，KH Coder に内蔵の R により得られた数値化された中心性得点を，SAS，JMP，SPSS 等に出力することも可能であり，テキストデータの分析をさらに深めることができる。

本稿の説明が，テキストデータ分析を検討中あるいは KH Coder の分析結果を用いてより高

度の解析を希望する大学生，大学院生および研究者の方々の参考になれば幸いである。

参考文献

[1] 樋口耕一, テキスト型データの計量的分析―2 つのアプローチの峻別と統合, 理論と方法, 19(1)，pp101-115，2004.
[2] 佐藤浩一. 自伝的記憶の機能と想起特性, 群馬大学教育学部紀要　人文・社会科学編, 56, pp 333-357，2007.
[3] 樋口耕一, KH Coder 2.x リファレンス・マニュアル, 2013. http://jaist.dl.sourceforge.net/project/khc/Tutorial/for KH Coder 2.x/khcoder_tutorial.pdf（平成 25 年 8 月 9 日閲覧）
[4] 田中彩佳，KH Coder を用いた自由記述データのテキスト分析，久留米大学情報教育センター，コンピュータジャーナル Vol.25，pp33-47，2010.
[5] 林俊克，Excel で学ぶテキストマイニング入門，オーム社，2002.
[6] 樋口耕一，KH Coder 掲示板（フォーラム），2013.　http://koichi.nihon.to/cgi-bin/bbs_khn/khcf.cgi　（平成 25 年 8 月 9 日閲覧）
[7] 鈴木努，R で学ぶサイエンス 8―ネットワーク分析―，共立出版，2009.
[8] Naoki Masuda & Norio Konno ，複雑ネットワーク　補遺（PDF 版），近代科学社，2010. http://www.stat.t.u-tokyo.ac.jp/~masuda/CN_supplement.pdf　（平成 25 年 10 月 31 日閲覧）
[9] 増田直紀，私たちはどうつながっているのか―ネットワークの科学を応用する―，中央公論新社，2007.