

qgraph によるネットワークの構築と描画

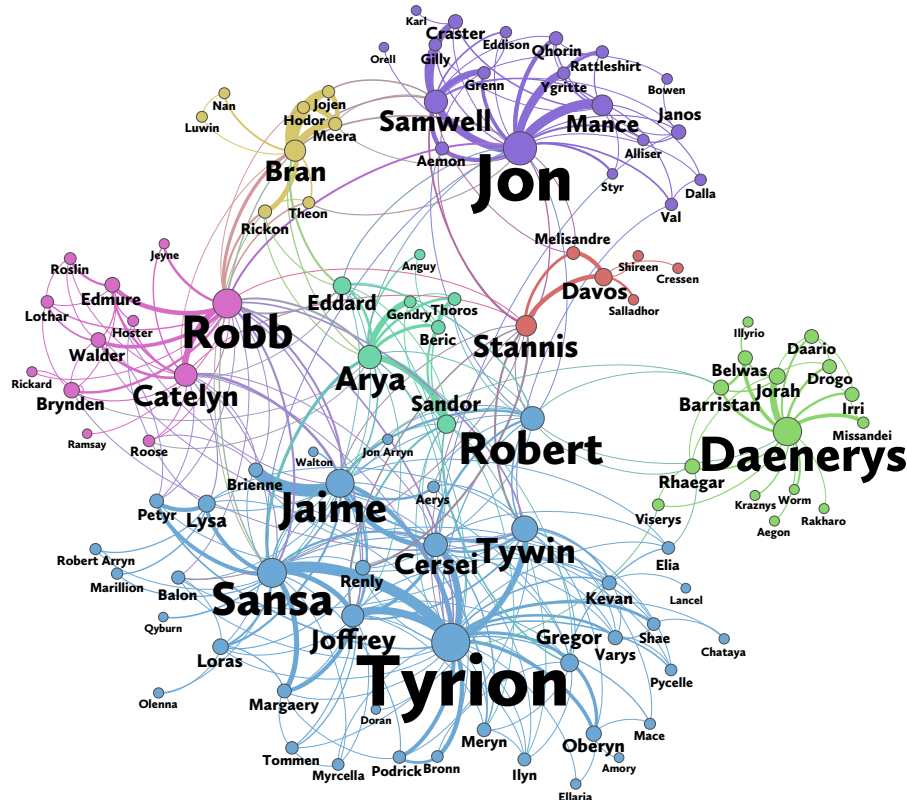
4 章の練習問題

アダラ＝マリア・イスヴォラヌ¹ &
サシャ・エプスカンプ^{1,2}

1. アムステルダム大学心理学科
2. アムステルダム大学都市精神健康センター

ネットワークの可視化

以下の練習問題では、シリーズ本**氷と炎の歌 (A Song of Ice and Fire)**の重みつき社会ネットワークについて分析する。この物語は、有名なテレビドラマシリーズ**ゲーム・オブ・スローンズ (Game of Thrones)**の原作となっている。このネットワークは、『**数学の地平**』誌 ([Math Horizons Magazine](#)) 上で公開されたものであり、テレビシリーズでいうところの第3,4シーズンにあたる、第三巻「**剣嵐の大地 (A Storm of Swords)**」に基づいたものである。データも含めたさらなる詳細については、[オンライン資料](#)を参照のこと。そこで公開されているネットワークは、以下の通りである。



Exercise 1 このネットワークが何を表しているのかを理解するために、Network of Thrones の Web サイト上で公開されている記事をざっと見てみよう。ノードとエッジは、それぞれ何を表しているだろうか。

以下のようにすれば、オンラインで公開されているデータを R に読み込むことができる (データファイルは、[本書の特設サイト](#)から入手できる)。

```
Data <- read.csv("stormofwords.csv")
```

Exercise 2 RStudio で View 関数を用い、データを眺めてみよう。この行列は、ひとつのネットワークを符号化したものである。どのようにして符号化されているかわかるだろうか。行は何を表し、列は何を表しているだろうか。

この構造は、ネットワークを符号化する**エッジリスト (edgelist)** として知られており、`qgraph` の入力として以下のように用いることもできる。

```
library("qgraph")
qgraph(Data, directed = FALSE)
```

Exercise 3 隣接行列や重み行列を入力として無向グラフをプロットする際には、通常は `directed` 引数を指定する必要がない。しかし、今回のようにエッジリストを用いる場合には、`directed` 引数を指定しなければならない。なぜそうなるのかを説明しなさい。

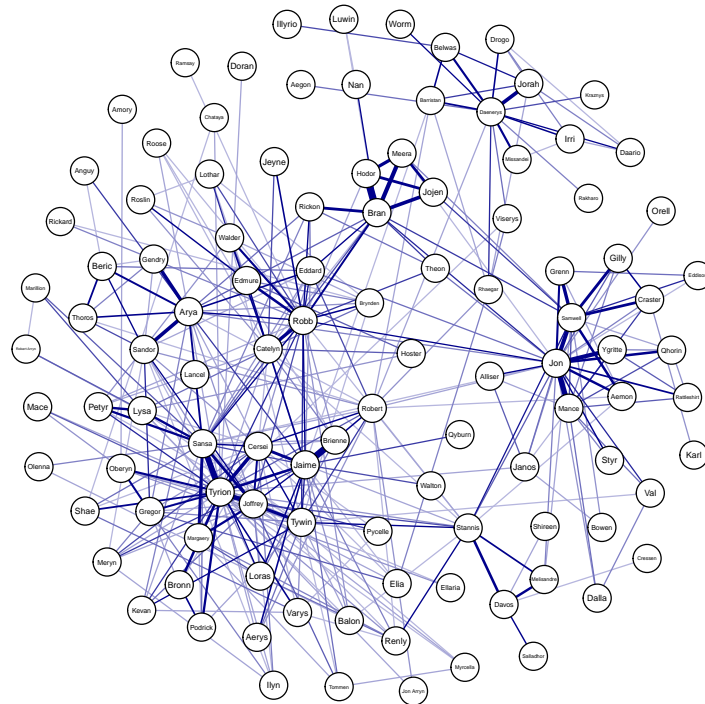
今度は、以下のデータセットを読み込もう。これも [本書の特設サイト](#) から入手できる。

```
Data2 <- read.csv("stormofwords_wmat.csv")
```

このデータセットは上記の同一のネットワークを表したもののだが、表現形式が異なっている。この表現形式は、**重み行列 (weights matrix)** と呼ばれている。

Exercise 4 `View(Data2)` を実行し、この新しいデータセットを精査してみよう。重み行列がどのようにしてネットワークを符号化しているかわかるだろうか。また、`Data2` を入力として、`qgraph()` を実行してみよう。上記と同一のネットワークは得られるだろうか。`directed` 引数は、今回も必要だっただろうか。

プロットされたネットワーク図は、円形レイアウトを用いて描画されている。しかし、今回のようにノードが多いネットワークの場合、この円形レイアウトでは解釈や読み取りが困難となってしまう。さらに、正の値のエッジを表示する際に、緑ではない別の色で表示したいということもあるだろう。また、ノードが非常に大きくなっているので、もっと小さくしたいということもあるだろう。そのような操作を実行すると、以下のようなネットワーク図が得られる。



Exercise 5 上記のプロットを自分でも作ってみよう。つまり、レイアウトをスプリングレイアウトに変更し、エッジの色を“darkblue”に変え、ノードのサイズを3に変更しよう。qgraph のヘルプページを参照し (?qgraph), 必要なコマンドを探してほしい。なお、あなたの手元のコンピュータでは、解答例とまた異なったスプリングレイアウトが出力される (つまり、ノードが異なった場所に配置される) かもしれない。

qgraph では、ネットワークの解釈に必要な3つの引数を用いている。それは、minimum, cut, maximum の3つである。これらの設定は自動的になされるが、引数で detrials = TRUE と指定すればその設定を表示できる。

Exercise 6 1つ前の問題において、qgraph では cut と maximum にどういった値が割り振られていただろうか。なお、minimum はデフォルトとして常に0と設定されている。そのため、0と異なる値が手動で設定されていない限りは、details = TRUE としても表示されない。

minimum 引数を用いれば、重みの絶対値が特定の値を下回るエッジを隠すことができる (つまり、負のエッジも正のエッジとして処理される)。ただし、それらのエッジは視覚的に隠されているだけであり、以降の分析で除外されているわけではないことに注意してほしい (以降の分析で除外したい場合には、threshold 引数を用いるとよい)。この引数は、密なグラフ (たとえば、相関ネットワーク) をプロットする際には便利だが、本書で推定するようなネットワークの場合には推奨されない。

Exercise 7 スプリングレイアウトを用いる際に、`minimum` 引数の設定を 1, 10, 20 と変えていってみよう。ネットワークはどのように変容するだろうか。また、同じことを今度は `threshold` 引数を用いてやってみよう。`minimum` を用いた際にはレイアウトが同一のままだったのに、`threshold` を用いた場合にはレイアウトが変わってしまう理由はなぜか、説明できるだろうか。

`qgraph` においては、エッジの重みの絶対値がより強く (大きく) なるにつれて、エッジがより太くかつより高い彩度で描かれるようになる。また、大きなネットワークの場合は、太さや色のスケールを途中で変更する (スプリットする) ことが有用だろう。`cut` 引数を用いられるのはこのためである。つまり、重みの絶対値がカットオフ値を下回るエッジであれば、最も細い線で描画され、色だけが変動するということになる。一方、重みの絶対値がカットオフ値を上回るエッジであれば、彩度が常に最大となり (色は、緑と赤がデフォルトである)、エッジが強くなるにつれてより太く描かれるようになる。なお、`cut = 0` と設定すれば、こうした挙動を無効化することができる。そうした場合には、いかなる場合でも、すべてのエッジが色と太さの両面でスケールに沿って変化するようになる。なお、`qgraph` では、20 個以上のノードを含むネットワークの場合にはカットオフ値が自動的に設定されるようになっている。しかし、そうした挙動が特別に必要だということでもない限りは、`cut = 0` と設定することを推奨しておく。

Exercise 8 スプリングレイアウトを用いる際に、`cut` 引数を 50, 10, 1 と変えていってみよう。ネットワークはどのように変容するだろうか。今度は、`cut` 引数を 0 に設定してみよう。何が起こるだろうか。

エッジのスケールリングは、そのネットワークに含まれる最も強い絶対値を持ったエッジを基準に選択される。なぜならば、エッジの重みはネットワークごとに大きく異なってくるからだ。上記の社会ネットワークの場合、エッジの重みはキャラクター間の相互作用の回数に基づいて決まっている。つまり、「ブラン (Bran)」と「ホーダー (Hodor)」というキャラクターが最も頻繁に交流しており (96 回)、最も強いエッジの重みが 96 になっているというわけだ。しかし (偏) 相関に基づくネットワークの場合は、最も強いエッジの重みが 1 を超えるということとはあり得ない (そして通常は、1 よりも弱いエッジが目立つものである)。ここで、`maximum` 引数を用いれば、エッジの色と太さのスケールリングの基準となる「エッジの最大値」を上書きすることができる。そこで指定した値は、見えないエッジの重みとして処理され、そのネットワークではその値こそが最大のエッジの重みとして設定される。`maximum` をより大きくしていくと、「ネットワーク内に実在する最も強いエッジ」の代わりに、`maximum` で指定した値を基準にエッジがスケールするようになる。

Exercise 9 `maximum` 引数を 200 に設定してみよう。何が起こるだろうか。今度は、`maximum` をデフォルトにいったん戻したうえで、10 に設定し直してみよう。`maximum = 200` としたときにはネットワークが変容したのに、`maximum = 10` としてもデフォルト値からネットワークが変容しないのはなぜか。

Exercise 10 (偏) 相関をもととしたネットワークを描く際には、`maximum` をどのように設定しておくのがよいだろうか。

Exercise 11 `qgraph` を用いて、このネットワーク構造についての中心性プロットを描画してほしい。プロットには強度、近接性、媒介性を必ず含め、 z 値は報告しないこと。ヒント：まずは、ネットワーク構造を個別のオブジェクトに保存しておく必要がある。