Samenvatting Introduction Computational Science – deel 2

Lucas Riedstra

27 maart 2019

Inhoudsopgave

1 Introductie

1.1 Gevaren

Er schuilt een gevaar in complexe systemen, namelijk dat fouten kunnen escaleren. Voorbeelden zijn de financiële crisis en stroomstoringen. Om dit te voorkomen moeten we de structuur het *netwerk* begrijpen achter het systeem, moeten we deze netwerken kunnen modelleren, en moeten we begrijpen hoe de netwerkstructuur dit soort fouten kan veroorzaken.

Een ander gevaar van netwerken is *kwetsbaarheid door interconnectiviteit* – als een netwerk sterk verbonden is, is het goed mogelijk dat lokale fouten niet lokaal blijven.

Deze gevaren vormen een reden om netwerken te willen bestuderen.

1.2 Netwerken bij complexe systemen

Complexe systemen zijn systemen wiens gedrag niet (makkelijk) afleidbaar is uit kennis van de losse componenten. Het zijn zelf-organiserende, evoluerende en dynamische systemen. Bij elk alle systemen horen netwerken, en het is onmogelijk een complex systeem te begrijpen zonder de netwerken erachter te begrijpen. Het netwerk definieert de interactie tussen de componenten.

De reden dat *network science* juist nu zo populair is, is tweeledig: ten eerste was er nooit genoeg opslag voor de data achter complexe netwerken of genoeg rekenkracht om iets te kunnen doen met die data. Ten tweede blijkt het dat de 'architecturen' achter netwerken in verschillende takken van de wetenschap op elkaar lijken, en dat dezelfde wiskundige technieken gebruikt kunnen worden om ze te bestuderen. Een derde reden is dat het *nodig* blijkt te zijn om complexiteit te begrijpen.

Network science wordt als volgt gekarakteriseerd:

- Ze is *interdisciplinair*;
- Ze is *empirisch* (en niet abstract zoals de wiskunde erachter);
- Er zit een sterk wiskundig formalisme achter;
- Er zit een sterke computationele component in, vooral door de gigantische probleemgroottes.

Het is bijna onnodig om te vermelden dat *network science* een gigantische impact heeft gehad, zowel in de samenleving als in de wetenschap.

2 Grafentheorie (AKA: veel definities)

Definitie 2.1. Bij elk netwerk hoort een *graaf* G bestaat uit een aantal *knopen* $1, 2, \ldots, N$, en *kanten* tussen die knopen. Het aantal kanten in een graaf noteren we met L. Het *maximale* aantal kanten in een graaf met N knopen noteren we met $L_{\max} = \binom{N}{2}$ (hierbij gaan we er vanuit dat self-loops niet toegestaan zijn). Als $L = L_{\max}$, noemen we G *compleet*.

De kanten kunnen zowel *gericht* als *ongericht* zijn – dit hangt af van het netwerk dat bestudeerd wordt. Zijn alle kanten in een graaf ongericht, dan noemen we de graaf zelf ongericht, en zijn alle kanten gericht, dan noemen we de graaf gericht.

We noemen G daarbij *gewogen* als we aan elke kant van G een getal ('gewicht') toekennen. We noemen G een *multigraaf* als er meerdere kanten mogelijk zijn tussen twee knopen (dit is hetzelfde als een gewogen graaf waarbij de gewichten enkel positieve getallen mogen zijn).

Natuurlijk is L in een echt netwerk (bijna) nooit gelijk aan $L_{\max}$, echte netwerken zijn helemaal niet dicht.

Natuurlijk zijn er meerdere manieren mogelijk om een netwerk te representeren als graaf.

Definitie 2.2. Zij $x_1, \ldots, x_N$ een steekproef (*sample*). Dan definiëren we de volgende grootheden:

- Het n-de moment:
$$\langle x^n \rangle = \frac{1}{N} \sum_{i=1}^{N} x_i^n.$$
Het eerste moment noemen we ook het *gemiddelde*.
- De *standaarddeviatie*:
$$\sigma_x = \sqrt{\frac{1}{N}(x_i - \langle x \rangle)^2}.$$

- De *distributie*:
$$p(x) = \frac{1}{N} \sum_{i=1}^{N} \delta_{k,x_i},$$
met $\delta_{ij} = 1$ als $i = j$ en 0 anders. Dus simpelweg is $p_x(k)$ het aantal keer dat k voorkomt in de steekproef.

Definitie 2.3. Als G ongericht is, noemen we het aantal knopen waarmee i verbonden is de *graad* van i, notatie k_i. Zien we de graden nu als steekproef van grootte N, dan zien we dat er altijd geldt $\langle k \rangle = \frac{2L}{N}$. Bij deze steekproef hoort ook een distributie $p(k)$, dit noemen we de *gradenverdeling* (*degree distribution*).

Als G gericht is, noemen we het aantal knopen waar i naartoe wijst de *uitgraad* van i, notatie k_i^{out}, en het aantal knopen dat naar i wijst de *ingraad* van i, notatie k_i^{in}. De gemiddelde ingraad $\langle k^{\text{in}} \rangle$ en gemiddelde uitgraad $\langle k^{\text{out}} \rangle$ zijn altijd hetzelfde, en gelijk aan $\frac{L}{N}$.

Definitie 2.4. Zij G een graaf (niet per se gericht of ongericht), dan definiëren we de *adjacency*-matrix als $(N \times N)$-matrix A met $A_{ij} = 1$ als er een kant is van i naar j, en $A_{ij} = 0$ anders.

De ingraad van i is nu simpelweg de som van alle getallen in *kolom* i van A, en de uitgraad is de som van alle getallen in *rij* i van A.

Als G een gewogen graaf is, zetten we niet $A_{ij} = 1$ als er een kant loopt van i naar j, maar geven we als getal het bijbehorende gewicht.

Opmerking. In de slides wordt de adjacency-matrix anders gedefinieerd dan in het boek (in het boek stellen ze $A_{ij} = 1$ als er een kant loopt van j naar i). Ik vind de manier van het boek vreemd.

Definitie 2.5. We noemen een graaf *bipartiet* als de knopen te splitsen zijn in twee (disjuncte) verzamelingen U en V, zodat elke kant van U naar V gaat of vice versa.

Een simpel voorbeeld van een bipartiete graaf is als de knopenverzameling bestaat uit acteurs $a_1, \ldots, a_k$ en films $f_1, \ldots, f_\ell$, waarbij we een kant leggen tussen a_i en f_j als acteur a_i in film f_j gespeeld heeft.

Definitie 2.6. Een *pad* in G is simpelweg een rij punten $i_0, \ldots, i_n$ zodanig dat er een kant loopt van i_0 naar i_1, van i_1 naar i_2, et cetera. De *afstand* $d(i, j)$ van i naar j definiëren we als de lengte van het kortste pad van i naar j. Is er geen pad, dan stellen we $d(i, j) = \infty$. Als tussen elk paar punten in G een pad bestaat noemen we G *verbonden*. Anders kunnen we G opdelen in verschillende *samenhangende componenten*: deelgrafen van G die samenhangend zijn, en waar geen links tussen zijn. De grootste component noemen we de *giant*-component.

In het geval van gerichte grafen maken we onderscheid tussen *zwak verbonden* (als we de richting van de kanten negeren) en *sterk verbonden* (als we de richting van de kanten meenemen). Als G een sterk verbonden component (SVC) heeft, definiëren we de *incomponent* als de verzameling knopen die de SVC kunnen bereiken, en de *uitcomponent* als de verzameling knopen die bereikt kan worden vanuit de SVC.

Een pad met hetzelfde begin- en eindpunt heet een *cykel*, een pad dat elke kant precies één keer gebruikt heet een *Eulerpad*, en een pad dat precies elke knoop één keer langsgaat heet een *Hamiltonpad*.

De *gemiddelde padlengte* $\langle d \rangle$ is het gemiddelde van alle afstanden tussen elk paar knopen. De *diameter* $d_{\max}$ is het langste kortste pad.

Als G verbonden is, maar er bestaat een kant zodanig dat als die kant weggehaald zou worden, G niet meer verbonden zou zijn, dan noemen we die kant een *brug*.

Stappenplan 2.7. Er zijn in het algemeen twee algoritmen om padlengtes te vinden: *Breadth-first search* (BFS) en *depth-first search* (DFS). Bij beide algoritmen beginnen we met een bepaalde knoop i_0. Bij BFS bekijken we vervolgens alle knopen waar i_0 aan verbonden is, geven we die afstand 1, en stoppen we die in een queue. Zo gaan we recursief verder.

Definitie 2.8. Als G ongericht is, definiëren we de *clusteringscoëfficiënt* van een knoop i als

$$C_i = \frac{e_i}{\binom{k_i}{2}} = \frac{2e_i}{k_i(k_i-1)},$$

waarbij e_i het aantal kanten is tussen de *buren* van i (waarbij we i zelf niet meerekenen). Natuurlijk definiëren we ook de gemiddelde clusteringscoëfficiënt $\langle C \rangle$.

Opmerking. In het boek wordt in plaats van e_i de letter L_i gebruikt. Aangezien we L al gebruiken voor het aantal kanten van de graaf, vind ik dit een zeer slechte keuze.

3 Verschillende soorten netwerken

3.1 Regelmatige netwerken

Een *regelmatige netwerk* is een netwerk waar er een bepaalde structuur is, die arbitrair vaak herhaald kan worden. In een regelmatig netwerk zijn grootheden als $\langle k \rangle, \langle C \rangle, \langle d \rangle$ vaak makkelijk uit te rekenen, evenals de gradenverdeling $p(k)$ (vaak hebben alle knopen dezelfde graad). Het belangrijkste 1D-voorbeeld is een *ring*, het belangrijkste 2D-voorbeeld een *rooster*. Dit soort reguliere netwerken noemen we *lattices* in het Engels (ik denk dat het Nederlandse woord *tralie* is, maar weet ik niet zeker). In het algemeen geldt dat voor een D-dimensionaal lattice, de gemiddelde afstand iets is van $O(N^{1/D})$ (specifiek in een ring dus $O(N)$ en in een rooster $O(\sqrt{N})$).

3.2 Random grafen (Erdös-Rényi-model)

3.2.1 Definitie, simpele resultaten

Totaal tegenovergesteld aan regelmatige netwerken zijn *random grafen*.

Definitie 3.1. Gegeven een $p \in [0,1]$ definiëren we $G(N,p)$ als een random graaf met N knopen, waarbij we tussen elk paar knopen met kans p een kant leggen. Het moge duidelijk zijn dat $G(N,p)$ níét uniek gedefinieerd is.

We kunnen een aantal dingen observeren over random grafen:

1. Gegeven één graaf met N knopen en L kanten, is de kans dat $G(N,p)$ deze graaf oplevert gelijk aan $p(G(N,p)) \coloneqq p^L \cdot (1-p)^{L_{\max}-L}$.

2. De kans dat $G(N,p)$ een graaf oplevert met L kanten voor een gegen L is nu gelijk aan $p(G(N,p))$ vermenigvuldigd met het aantal mogelijkheden voor een graaf met L kanten en N knopen. Dit is gelijk aan $\binom{\binom{N}{2}}{L}$, dus de kant op een graaf met L knopen is nu

$$p(L) \coloneqq \binom{\binom{N}{2}}{L} p^L (1-p)^{L_{\max}-L}.$$

3. Bovenstaande formule geeft exact de kansmassafunctie van een binomiale verdeling met $n = \binom{N}{2}$ (de letters worden steeds verwarrender). We kunnen $G(N,p)$ zien als een binomiaal experiment, waarbij we $n = \binom{N}{2}$ keer een experiment uitvoeren (in dit geval 'het leggen van een kant'), dat succes heeft met kans p.

   Met deze kennis kunnen we alles gebruiken wat we al weten over binomial verdelingen en dit toepassen op random grafen: het *verwachte* aantal kanten in $G(N,p)$ wordt gegeven door $\langle L \rangle = p \cdot L_{\max}$. We wisten al dat $\langle k \rangle = 2L/N$ in een gegeven graaf, dus gemiddeld zal deze graad nu $2\langle L \rangle / N = p(N-1)$ worden.

3.2.2 Gradenverdeling, benadering met Poisson

Definitie 3.2. Bij een random graaf definiëren we de gradenverdeling iets anders: we kijken naar de kans dat een gegeven knoop graad k heeft. Dit kunnen we ook zien als een binomiaal experiment: we gaan alle andere $N-1$ knopen langs, en bij elk van die knopen ligt met kans p een kant. Als de graad k is, dan is er dus k keer wél een kant neergelegd en de andere $(N-1)-k$ keer niet. Dit geeft de verdeling

$$p(k) = \binom{N-1}{k} p^k (1-p)^{N-1-k}.$$

Gezien dit ook een binomiale verdeling is, deze keer met $n = N-1$, kunnen we hiervan ook de verwachting uitrekenen (dit wordt $p(n-1)$) en de standaarddeviatie (dit wordt $\sqrt{p(1-p)(N-1)}$).

We zijn nu, zowel in de kantenverdeling als in de gradenverdeling, een binomiale verdeling tegengekomen. Deze heeft een nadeel: er zijn twee parameters, n en p. Er is een verdeling die de binomiale verdeling goed benadert: de zogenaamde *Poissonverdeling*. Deze wordt gegeven door

$$p(k) = e^{-\lambda} \frac{\lambda^k}{k!},$$

met λ een of andere positieve parameter. Als we voor λ nu $\langle k \rangle$ kiezen, blijkt dit een goede benadering voor de gradenverdeling te zijn, mits N en p groot genoeg zijn. Voor een Poissonverdeling wordt het gemiddelde gegeven door λ (dus in ons geval $\langle k \rangle$), en de standaarddeviatie wordt gegeven door $\sqrt{\lambda}$, dus $\sqrt{\langle k \rangle}$ in ons geval.

Merk op dat bij benadering door een Poissonverdeling het aantal knopen *niet* uitmaakt, alleen de gemiddelde graad. Dit impliceert dat random netwerken met dezelfde gemiddelde graad dezelfde structuur zullen hebben (in ieder geval voor $N \to \infty$).

Een enigszins verdrietig resultaat is dat *real-life*-netwerken bijna nooit Poisson-netwerken zijn, omdat de Poissonverdeling geen 'dikke staarten' heeft: outliers zijn praktisch gezien nonexistent bij Poissonverdelingen, terwijl de meeste real-life-netwerken wel outliers hebben.

3.2.3 De *giant* component

Herinner dat de giant component de *grootste* samenhangende component van een graaf is. We noteren het aantal knopen in deze component met N_G. Duidelijk geldt voor $p = 0$ dat $N_G/N = 0$ en voor $p = 1$ dat $N_G/N = 1$. We vragen ons nu af hoe deze stijging eruit ziet.

Het blijkt dat, wanneer $\langle k \rangle > 1$ (ook al is het maar een klein beetje groter), er een giant component ontstaat, en N_G/N gigantisch snel naar 1 convergeert. Dit is equivalent met $p = \frac{1}{N-1} \approx \frac{1}{N}$. Hiermee kunnen we het netwerk indelen in vier klasses:

1. Het *subkritieke* gebied: $0 < \langle k \rangle < 1$. Als de gemiddelde graad van een knoop lager dan 1 is, zal er geen giant component ontstaan, en zal gelden $N_G/N \approx 0$. Dit soort netwerken bestaan uit kleine componentjes met groottes die exponentieel verdeeld zijn.

2. Het *kritieke* gebied: $\langle k \rangle = 1$. Het krieke gebied scheidt het subkritieke gebied van superkritieke gebied. In het kritieke gebied hebben we nog steeds kleine componentjes, maar hun groottes zijn nu verdeeld volgende de *power law*.

3. Het *superkritieke gebied*: $\langle k \rangle > 1$. Voor k iets groter dan 1 (maar niet te veel), geldt $N_G/N \approx \langle k \rangle - 1$. We hebben nu één giant component en een aantal andere knopen die verdeeld zijn als

   $$p(s) \approx s^{-3/2} e^{(-\langle k \rangle - 1)s + (s-1)\ln k},$$

   met s de kans voor een knoop om in een component van grootte s te zitten.

4. Het *verbonden* gebied: $\langle k \rangle > \ln N$. Op een gegeven moment zal de giant component (bijna) alle knopen in het netwerk bevatten. Dit blijkt te gebeuren rond $\langle k \rangle \approx \ln N$. Dan geldt dus $N_G/N \approx 1$.

Het blijkt dat bijna alle real-life-netwerken meestal in het superkritieke gebied liggen (internet, proteïne-interacties), en sommige in het verbonden gebied (acteurs en films).

3.2.4 Gemiddelde afstanden (random netwerk ≈ boom)

Gegeven een random graaf met gemiddelde graad $\langle k \rangle$, en een of andere vast gekozen knoop i, zal deze $\langle k \rangle$ knopen hebben op afstand 1, $\langle k \rangle^2$ knopen op afstand 2, etc., tot en met $\langle k \rangle^d$ knopen op afstand d. Dus het aantal knopen op afstand maximaal d van i wordt gegeven door $N(d) \coloneqq 1 + \langle k \rangle + \langle k \rangle^2 + \cdots + \langle k \rangle^d$, en we weten dat dit gelijk is aan $\frac{\langle k \rangle^{d+1}-1}{\langle k \rangle - 1}$. Kiezen we nu $\langle k \rangle >> 1$, dan kunnen we de (-1)-termen weggooien en vinden we $N(d) \approx \langle k \rangle^d$.

Voor de maximale afstand moet gelden $N(d_{\max}) \approx N$ (want alle knopen hebben afstand $d_{\max}$ of minder van i). En we zien dus

$$N(d_{\max}) \approx N \implies \langle k \rangle^{d_{\max}} \approx N \implies d_{\max} \approx \log_{\langle k \rangle}(N).$$

In realiteit blijkt deze formule de gemiddelde graad $\langle d \rangle$ beter te benaderen dan $d_{\max}$. We schrijven dus

$$\langle d \rangle \approx \log_{\langle k \rangle}(N),$$

en dit noemen we de *small-world*-eigenschap: de gemiddelde 'afstand' tussen twee personen op aarde is vele ordes van grootte kleiner dan het aantal mensen N. Dit komt vooral omdat $\langle k \rangle$ voor de meeste mensen nogal groot is.

Merk op dat dit i.h.a. een veel kleiner resultaat geeft dan de formule $\langle d \rangle \approx N^{1/D}$ die we hadden afgeleid voor reguliere netwerken.

3.2.5 Gemiddelde clusteringscoëfficiënt

Voor de gemiddelde clusteringscoëfficiënt van een knoop i moeten we een schatting hebben voor het aantal links tussen de buren van die knoop. Gezien er $\binom{k_i}{2}$,mogelijkheden zijn en we ze kiezen met kans p, is het verwachte aantal links $e_i = p \cdot \binom{k_i}{2}$. Gezien een knoop gemiddeld graad $\langle k \rangle$ heeft, geeft dit als formule voor de gemiddelde clusteringscoëfficiënt

$$\langle C \rangle = \frac{2e_i}{\langle k \rangle (\langle k \rangle - 1)} = p \cdot \binom{\langle k \rangle}{2} \Big/ \binom{\langle k \rangle}{2} = p.$$

Het blijkt dat random netwerken de clusteringscoëfficiënt *zeer* onderschatten: random netwerken geven een soort 'minimale' $\langle C \rangle$, maar in realiteit zal deze veel hoger liggen. Ook bestaan random netwerken veel meer uit losse clusters.

3.2.6 Conclusie

Random netwerken benaderen de gemiddelde afstand $\langle d \rangle$ goed, maar de gradenverdeling $p(k)$ en de clusteringscoëfficiënt $\langle C \rangle$ slecht. In het algemeen is een random netwerk nooit de goede keuze voor een model.

Waarom bestuderen we dan toch random netwerken? Om een dieper begrip voor netwerken te krijgen, en om andere netwerken mee te vergelijken. Als een eigenschap van random netwerken voorkomt in een real-life-netwerk, kan dat duiden op willekeur, en als een eigenschap afwezig is, kan dat duiden op een zekere structuur.

3.3 Watts-Strogatz-model

Het Watts-Strogatz model combineert *regulariteit* (lokale clusters, $\langle d \rangle$ groot), met *willekeur* (weinig clustering, kleine $\langle d \rangle$). Dit wordt als volgt gedaan: kies een of andere $p \in [0, 1]$, en begin met een regulier netwerk (in het 1D-geval: een ring). Ga vervolgens élke kant langs, en vervang deze met kans p door een willekeurige andere kant.

Voor $p = 0$ houden we het reguliere model, en voor $p = 1$ krijgen we een random graaf $G(N, L/L_{\max})$. Het interessante is natuurlijk wat daartussenin gebeurt. Voor het voorbeeld $N = 1000, \langle k \rangle = 10$ zien we het volgende: de geiddelde clusteringscoëfficiënt blijft een tijdje heel hoog blijft, en begint dan rond $p \approx 0.1$ hard te dalen. Ook daalt de gemiddelde afstand tot $p \approx 0.1$ best hard tot bijna nul (ongeveer 0.05), en blijft daarna laag.

We zien dus dat we voor $p \approx 0.1$ een model hebben met een hoge clusteringscoëfficiënt (lokaliteit), én een lage gemiddelde afstand (globaliteit). Dit geeft dus al een iets betere benadering van real-life-netwerken.

3.4 Schaalvrij netwerk (*scale-free*)

We zagen al eerder dat de Poissonverdeling een slechte benadering is voor de gradenverdeling van real-life-netwerken omdat real-life-netwerken meer outliers hebben. Ze blijken beter benaderd door de volgende verdeling:

$$p(k) = C \cdot k^{-\gamma}, \qquad (\text{“}\textit{power law}\text{”})$$

voor zekere $\gamma > 0, C > 0$. We noemen γ de *gradenexponent* (*degree exponent*). Merk opdat we nu uitgaan van een *continu* model, waarbij graden in theorie dus ook niet-geheel kunnen zijn. Dit maakt de berekeningen makkelijker.

Met zo'n model vinden we nu dat de kans dat de graad van een knoop tussen k_1 en k_2 ligt, gelijk is aan

$$\int_{k_1}^{k_2} p(k)\,\mathrm{d}k.$$

Definitie 3.3. Een *schaalvrij netwerk* is een netwerk waarvan de gradenverdeling een *power law* is.

Het verschil tussen een *power law*-verdeling en de vorige (binomiale/Poisson/exponentiële) verdelingen is dat de kans op uitschieters bij de power law hoger is: er zal een klein aantal nodes zijn met hele hoge graad (de zgn. *hubs*).

We weten dat echte netwerken eindig zijn, en dus hebben we een maximale graad $k_{\max}$. Natuurlijk willen we dat $\mathrm{P}(k > k_{\max}) = 0$, maar we zitten in een continu model, dus het volstaat dat $\mathrm{P}(k > k_{\max}) = \frac{1}{N}$.

We berekenen ook de C in de verdeling: er moet gelden $\mathrm{P}(k \geq k_{\min}) = 1$, en dus

$$C \int_{k_{\min}}^{\infty} k^{-\gamma}\mathrm{d}k = 1 \implies \frac{C}{1-\gamma} \cdot -k_{\min}^{1-\gamma} = 1 \implies C = (\gamma - 1) \cdot k_{\min}^{\gamma-1}.$$

We hebben dus

$$\frac{1}{N} = \mathrm{P}(k > k_{\max}) = (\gamma - 1) K_{\min}^{\gamma-1} \int_{k_{\max}}^{\infty} k^{-\gamma\,\mathrm{d}k} = \left(\frac{k_{\min}}{k_{\max}}\right)^{\gamma-1}.$$

Hieruit kunnen we afleiden dat

$$k_{\max} = k_{\min} \cdot N^{\frac{1}{\gamma-1}}.$$

De grootte van de grootste *hubs* schalen mee met de grootte van het netwerk. Voor exponentiële verdelingen kunnen we een analoge berekening doen (met $p(k) = Ce^{-\lambda k}$), dus eerst C berekenen en dan $\mathrm{P}(k > k_{\max})$ uitrekenen, dit geeft $k_{\max} = k_{\min} + \frac{\ln N}{\lambda}$.

We zien dus dat voor een exponentiële verdeling, $k_{\max}$ en $k_{\min}$ relatief dichtbij elkaar liggen, terwijl bij een *power law*-verdeling, deze ver weg van elkaar zullen zijn.

Een zeer interessante conclusie is dat voor *real-life*-netwerken, er bijna altijd geldt dat γ tussen 2 en 3 ligt. In dit geval is de verwachting $\langle k \rangle$ eindig, maar de variantie/standaarddeviatie oneindig (het netwerk is *scale-free*).

We kunnen de scale-free-netwerken nu in groepen verdelen:

1. $\gamma = 2$: gemiddelde afstand $\langle d \rangle$ is constant;
2. $2 < \gamma < 3$: *ultra-small-world*: gemiddelde afstand $\langle d \rangle \approx \ln \ln N$ (nee, geen typfout, twee keer ln), dus heel klein.
3. Kritiek punt $\gamma = 3$, $\langle d \rangle = \frac{\ln N}{\ln \ln N}$.
4. Small world $\gamma > 3$, $\langle d \rangle = \ln N$. Een belangrijke opmerking $\gamma > 3$, de variantie eindig is. Het netwerk is nu dus niet meer scale-free en lijkt meer op een random netwerk.

Om meer te begrijpen over deze netwerken moeten we kijken naar de momenten. Het n-de moment wordt gegeven door

$$\langle k^n \rangle \approx \int_{k_{\min}}^{\infty} k^n p(k) \, \mathrm{d}k.$$

Bij een schaalvrij netwerk is dit

$$\langle k^n \rangle = C \cdot \frac{k_{\max}^{n-\gamma-1} - k_{\min}^{n-\gamma-1}}{n - \gamma - 1}.$$

We zien dus dat alle momenten met $n \leq \gamma - 1$ eindig zijn, en alle momenten met $n > \gamma - 1$ divergeren naar oneindig. Voor $\gamma \in (2, 3)$ is het eerste moment, het gemiddelde, dus eindig, maar het tweede moment is oneindig, en de variantie dus ook. **Dit is waarom dit netwerk *scale-free* genoemd wordt**. De variantie is oneindig, dus voor een willekeurig gekozen knoop heb je geen idee wat de graad is.

Voor $\gamma > 3$ zal het netwerk meer op een random netwerk lijken (eindig gemiddelde, eindige variantie). Voor $\gamma < 1$ divergeren zowel het gemiddelde als de variantie. Er zijn bijna geen *real-life* netwerken in dit gebied.

3.5 Barabasi-Albert-model

In al onze vorige modellen was het aantal knopen N constant. Het BA-model (Barabasi-Albert) varieert dit. Bij het toevoegen van knopen hebben we twee opties:

- Ofwel de nieuwe knopen linken volledig willekeurig aan een aantal bestaande knopen;
- Ofwel de nieuwe knopen hebben een voorkeur om te linken aan knopen die al heel groot zijn. Dit noemen we *preferential attachment* of het *rich gets richer*-effect.

Het BA-model gebruikt de optie van *preferential attachment*: elke tijdstap introduceren we één nieuwe knoop die m nieuwe links maakt. De kans dat een nieuwe knoop linkt aan knoop i wordt gegeven door (we gebruiken nu ineens Π voor kansen):

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j},$$

dus grotere graad betekent grotere verbindingskans. Als we beginnen met m_0 knopen in ons netwerk en ℓ_0 kanten tussen die knopen, krijgen we dus $N(t) = t + m_0$ knopen en $L(t) = \ell_0 + mt$ kanten.

De gradenverdeling die tevoorschijn komt is een *power law*-verdeling met $\gamma \approx 3$. Deze γ is onafhankelijk van m, het aantal knopen dat we aan het begin toevoegen. De coëfficiënt C is proportioneel aan m^2.

3.5.1 Wiskundig onderzoek: groei van de graden

We onderzoeken nu hoe de graad van een knoop afhangt van de tijd: Stel dat een knoop graad k_i heeft. Dan zal deze er in een tijdstap gemiideld $m \cdot \Pi(k_i)$ buren bij krijgen, en dit benaderen met een differentiaalvergelijking geeft

$$\frac{\mathrm{d}k_i}{\mathrm{d}t} = m \cdot \Pi(k_i) = m\frac{k_i}{\sum_j k_j}.$$

De onderste som gaat over *alle* knopen in het netwerk behalve de nieuw toegevoegde knoop, en wordt dus gegeven door $2mt - m$ (de som van de graden is twee keer het aantal kanten):

$$\sum_j k_j = 2mt - m \implies \frac{\mathrm{d}k_i}{\mathrm{d}t} = \frac{k_i}{2t-1} \approx \frac{1}{2}\frac{k_i}{t} \implies \frac{\mathrm{d}k_i}{k_i} \approx \frac{1}{2}\frac{\mathrm{d}t}{t},$$

en nu links en rechts integreren geeft

$$\ln k_i = \frac{1}{2}\ln t + C \implies k_i = C \cdot t^{1/2},$$

en aangezien we weten dat $k_i(t_i) = m$ krijgen we $C = \frac{m}{t_i^{1/2}}$ en dus

$$k_i = m \cdot \left(\frac{t}{t_i}\right)^{1/2}$$

De exponent $1/2$ wordt ook genoteerd met β, de zogenaamde *dynamische exponent*. We zien dus dat de graden sublineair groeien, en dat een knoop dus gemiddeld steeds minder nieuwe buren krijgt.

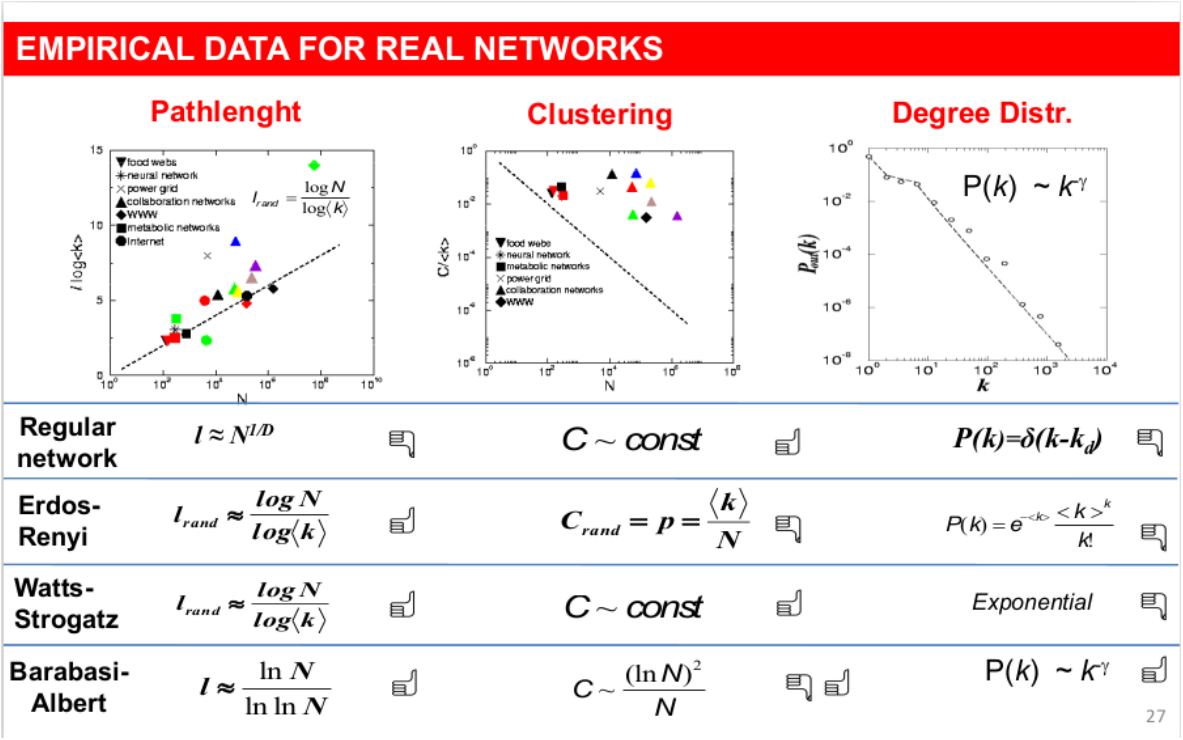
3.5.2 Wat gaat goed?

De gradenverdeling is in het BA-model dus een power law-verdeling met $\gamma = 3$, en dit is realistisch voor real-life-netwerken. We wisten al dat $\gamma = 3$ impliceert dat de gemiddelde afstand tussen knopen proportioneel is aan $\ln N / \ln\ln N$. Dit komt overeen met het *small-world*-model (kleine afstanden), dus ook dit wordt goed gerepresenteerd door het BA-model. Er is ook door anderen berekend dat de clusteringscoëfficiënt van de vorm

$$C = \frac{m}{8}\frac{\ln^2 N}{N}$$

is. Dit is lagere clustering dan we in *real-life*-modellen zien, en de clustering wordt ook steeds lager, terwijl we normaal juist constante clustering zien. Het BA-model benadert clustering dus redelijk, maar niet perfect.

Alles met alles geeft dit de volgende vergelijking van de besproken modellen:



Figuur 1: Alle data voor verschillende modellen, gestolen van de slides.

3.5.3 *Preferential attachment*: hoezo?

Er zijn twee soorten *preferential attachment*:

- Lokaal: mechanismes die geen kennis hebben van het hele netwerk en alleen lokaal kijken. Een voorbeeld: voor een knoop die we toevoegen met m plekken voor nieuwe links, kiezen we een willekeurige kant in het systeem. Bij die kant voegen we de knoop toe aan de kant die de hoogste graad van de twee punten heeft. Dit herhalen we dan voor de kanten die aan de nieuw verbonden knoop zitten.

  Real-life-voorbeeld: sociale netwerken.

- Globaal: mechanismes die bij plaatsing van nieuwe knopen / kanten het volledige netwerk meenemen in de beslissingen, zoals het plaatsen van nieuwe internetkabels.

Natuurlijk kan het BA-model worden geoptimaliseerd met bijvoorbeeld het vervangen/verplaatsen van kanten of het verplaatsen/verwijderen van knopen.

3.6 Centraliteit

Er zijn verschillende manieren om te beschrijven hoe *centraal* een bepaalde knoop is. Deze graden zijn altijd genormaliseerd om tussen 0 en 1 te liggen.

OPMERKING: In de slides wordt bij dit stuk ineens compleet nieuwe notatie gebruikt voor dingen die we al gezien hebben. Ik heb geen idee waarom, maar ik hou in ieder geval de oude notatie aan.

1. De meest simpele centraliteitsmeter is de graad van een knoop, gedeeld door de maximale graad. Dit heet de *degree centrality*, en geeft de 'populariteit' van een knoop weer. De degree centrality

van i is nu

$$\frac{k_i}{N-1}.$$

2. Een andere centraliteit is *closeness centrality*: de som van alle afstanden tot andere knopen, maar dan geïnverteerd (dus grotere afstand betekent lagere centraliteit). De closeness centrality van i is nu.

$$\frac{N-1}{\sum_{i\neq j} d_{ij}},$$

met d_{ij} de afstand tussen knoop i en j. We vermenigvuldigen nog met $N-1$ zodat als een knoop direct aan alle andere knopen verbonden is, deze closeness centrality 1 heeft.

Deze centraliteit geeft aan hoe snel informatie van deze knoop naar andere knopen kan komen.

3. Een andere belangrijke centraliteit is *betweenness centrality*: deze geeft aan hoe centraal deze knoop *tussen andere* knopen ligt, en dus hoe goed deze knoop informatie kan manipuleren. Dit wordt in twee stappen berekend voor een knoop i:

   - Eerst bereken je voor elk paar knopen j, k ongelijk aan i het **aantal** kortste paden $p(j,k)$. Dan bereken je het aantal kortste paden dat door i gaat, notatie $p_i(j,k)$. Vervolgens sommeer je over alle $p(j,k)/p_i(j,k)$, dus dit geeft

$$B(i) = \sum_{j,k\neq i} \frac{p_i(j,k)}{p(j,k)}$$

   - De zojuist gevonden waarde wordt gedeeld door het maximum van alle *betweenness centralities*, dus

$$\frac{B(i)}{\max_j B(j)}.$$

   - De laatste is *eigenvector centrality*: je berekent simpelweg de eigenvector bij de grootste eigenwaarde met grootste entry 1 van de *adjacency matrix*. Hoewel dit makkelijk klinkt, is het vinden van eigenwaardes en eigenvectoren voor matrices groter dan 4×4 in het algemeen een moeilijk probleem.

     Dit correspondeert met de *kans* dat een knoop informatie krijgt.

3.7 Robuustheid

Bij robuustheid onderzoeken we hoe bestend een netwerk is tegen het weghalen van knopen. Specifiek onderzoeken we de kans om bij de *giant component* te horen als deze er is.

Als we willekeurig knopen weghalen, blijken reguliere netwerken niet heel robuust te zijn: de grootte van N_G daalt snel, en al snel is er geen giant component meer.

Voor schaalvrije netwerken is het tegenovergestelde waar: ze blijken erg robuust. Dit komt omdat zo'n netwerk een aantal *hubs* heeft (knopen met hele hoge graad), en de kans om zo'n hub te verwijderen is erg klein omdat er zo weinig zijn.

De fractie knopen die verwijderd moet worden voordat de giant component verdwijnt noteren we met f_c (c voor *critical*). In het algemeen blijkt de formule

$$f_c = 1 - \frac{1}{\varkappa - 1}, \qquad \varkappa = \frac{\langle k^2\rangle}{\langle k\rangle}$$

te gelden. Er geldt dus: hoe hoger $\varkappa$, hoe hoger f_c. Definiëren we $r = |\frac{2-\gamma}{3-\gamma}|$, dan blijkt te gelden

$$\varkappa = \begin{cases} r \cdot K_{\min} & \text{als } \gamma > 3; \\ r \cdot K_{\max}^{3-\gamma} K_{\min}^{2-\gamma} & \text{als } \gamma \in (2,3); \\ r \cdot K_{\max} & \text{als } \gamma \in (1,2). \end{cases}$$

De 'zwakte' van schaalvrije netwerken is wel het volgende: we gaan nu uit van een *willekeurige* verwijdering van knopen. Zodra er *gerichte* aanvallen uitgevoerd worden op de hubs, valt een schaalvrij netwerk veel sneller uit elkaar dan een random netwerk. De oplossing is *backups*: meer hubs.

3.8 Epidemieën

3.8.1 Simpele modellen

Definitie 3.4. We definiëren een *epidemie* als een verspreiding van *iets* in een netwerk. Dit kan bijvoorbeeld een ziekte zijn onder mensen, een virus onder computers, of een gerucht op sociale media.

Er zijn meerdere modellen voor epidemieën, het simpelste is het 'SI' (*susceptible-infected*)-model: we hebben een vaste populatie N met daarin een aantal *susceptibles* en een aantal *infected*, waarbij mensen wel van S naar I kunnen gaan, maar niet terug. We krijgen dan een differentiaalvergelijking van de vorm

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta SI = \beta I(1-I).$$

Merk op dat we maar één differentiaalvergelijking nodig hebben: als we I hebben, weten we S ook, want $S = N - I$. De oplossing hiervan wordt iets van

$$I(t) = \frac{I_0 e^{\beta t}}{1 - I_0 + I_0 e^{\beta t}}.$$

In het begin is deze groei exponentieel, want dan geldt $I \approx 0$ en dus $\mathrm{d}I/\mathrm{d}t \approx \beta I$.

We kunnen het model iets ingewikkelder maken door te stellen dat geïnfecteerden (I) ook weer *susceptible* (S) kunnen worden. We krijgen dan

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta SI - \mu I = (\beta - \mu - \beta I)I.$$

De oplossing wordt nu gegeven door

$$I(t) = (1 - \mu/\beta)\frac{Ce^{(\beta-\mu)t}}{1 + Ce^{(\beta-\mu)t}}.$$

De factor β/μ noteren we ook met λ en noemen we de *reproductive number*: dit geeft weer hoeveel mensen ziek worden tegenover hoeveel mensen beter worden. Duidelijk stijgt het aantal geïnfecteerden als $\lambda > 1$ en daalt het als $\lambda < 1$. Het percentage geïnfecteerden zal ook convergeren naar λ.

Een laatste model is het SIR-model, waarin mensen ook beter (*recovered*) kunnen worden.

3.8.2 In netwerken

Opmerking. In de slides staan hier alleen maar een zooi formules zonder enige uitleg en in college is ze er ook doorheen gezoeft, dus I don't really know.

We bekijken eerst het SI-model: De factor

$$\tau := \frac{\langle k \rangle}{\beta(\langle k^2 \rangle - \langle k \rangle)}$$

geeft de tijd weer die het kost voor een epidemie om te groeien.

In het random netwerk geldt $\tau = (\beta\langle k \rangle)^{-1}$, dus hoe groter $\langle k \rangle$, hoe kleiner τ.

In het SF-model (scale-free) geldt $\langle k^2 \rangle \to \infty$ voor $N \to \infty$, dus dan ook $\tau \to 0$. Hieruit volgt dat een epidemie zich veel sneller kan verspreiden door een schaalvrij netwerk.

Voor het SIR-model geldt

$$\tau = \frac{\langle k \rangle}{\beta\langle k^2 \rangle - (\mu + \beta)k},$$

en wordt dit bij random netwerken gegeven door $(\beta\langle k\rangle - \mu)^{-1}$, en bij SF-netwerken gaat het nog steeds naar 0.

Met deze uitdrukkingen kunnen we een *kritieke* λ, λ_c afleiden, waarvoor een epidemie uitbreekt. In het SIR-model wordt deze λ_c in het random netwerk gegeven door $1/\langle k\rangle$, en in het schaalvrije model door

$$\lambda_c = \frac{1}{\frac{\langle k^2\rangle}{\langle k\rangle} - 1}.$$

3.9 Wiskundige afleidingen (kunnen op toets komen!)

3.9.1 Binomiale verdeling is ongeveer Poisson

We beginnen met een binomiale verdeling met $n = N - 1$ en $p \in [0, 1]$ (zoals bij de gradenverdeling). Dit geeft als kansmassa

$$p(k) = \binom{N-1}{k} p^k (1-p)^{N-1-k}.$$

Ervanuit gaande dat $k << N$, maakt het niet uit of we de trekking met of zonder terugleggen doen, dus krijgen we $\binom{N-1}{k} \approx \frac{(N-1)^k}{k!}$. We krijgen dus

$$p(k) \approx \frac{(N-1)^k}{k!} p^k (1-p)^{N-1-k}.$$

Nu willen we de term $(1-p)^{N-1-k}$ makkelijker schrijven. Merk hiervoor op dat

$$\ln\big((1-p)^{N-1-k}\big) = (N-1-k)\ln(1-p) = (N-1-k)\ln(1 - \frac{\langle k\rangle}{N-1})$$

De Taylorreeks van $\ln(1+x)$ op één term afschatten geeft $\ln(1+x) \approx x$ voor $|x| \leq 1$, en dus krijgen we

$$(N-1-k)\ln(1 - \frac{\langle k\rangle}{N-1}) \approx -(N-1-k)\frac{\langle k\rangle}{N-1} \approx -\frac{N-1-k}{N-1} \cdot \langle k\rangle \approx -\langle k\rangle,$$

want we namen aan dat $k << N$. Dus $\ln((1-p)^{N-1-k}) \approx -\langle k\rangle$, en dus $(1-p)^{N-1-k} \approx e^{-\langle k\rangle}$. Dit geeft als gradenverdeling

$$p(k) \approx \frac{(N-1)^k}{k!} p^k e^{-\langle k\rangle} = \frac{(N-1)^k}{k!} \cdot \frac{\langle k\rangle^k}{(N-1)^k} \cdot e^{-\langle k\rangle} = \frac{\langle k\rangle^k}{k!} \cdot e^{-\langle k\rangle}.$$

En dit geeft precies de verdeling die we wilden afleiden. De grote aanname was hier dus dat $k << N$, anders werkt dit ook echt niet.

3.9.2 De transitie van de giant component ligt bij $\langle k\rangle = 1$

Het goede nieuws is dat deze afleiding sterk lijkt op de vorige. Schrijf $S = N_G/N$, de fractie knopen in de giant component, en schrijf $u = 1 - S$, de fractie knopen níét in de giant component.

Bekijk een knoop i. De kans dat i níét deel is van N_G, wordt natuurlijk gegeven door u. Maar we kunnen deze kans ook anders opschrijven: als i níét deel is van N_G, geldt voor elke andere knoop dat deze andere knoop ofwel ook niet in N_G zit (maar dat i wel gelinkt is aan die knoop), ofwel dat er geen link tussen i en deze knoop bestaat. De kans op de eerste gebeurtenis is pu, de kant op de tweede is $1 - p$. Gezien er $N - 1$ andere knopen zijn geeft dit

$$u = (1 - p + pu)^{N-1}.$$

De rechterkant schrijven we weer kort met de taylorreeks van $\ln(1+x)$ zoals eerder, en we vinden

$$\begin{aligned}\ln(1-p+pu)^{N-1} &= (N-1)\ln(1-(1-u)p) = (N-1)\ln\left(1-(1-u)\frac{\langle k\rangle}{N-1}\right)\\ &\approx -(N-1)(1-u)\frac{\langle k\rangle}{N-1} = -(1-u)\langle k\rangle\\ &\implies (1-p+pu)^{N-1} \approx e^{-(1-u)\langle k\rangle} = e^{-\langle k\rangle S}.\end{aligned}$$

We hebben nu dus de vergelijking

$$u = e^{-\langle k\rangle S} \implies S = 1 - e^{-\langle k\rangle S}.$$

Om nu de precieze plek van de fasetransitie te vinden, moet ook gelden dat de afgeleides links en rechts gelijk zijn, dit geeft $\langle k\rangle e^{-\langle k\rangle S} = 1$. Vullen we nu $S = 0$ in (we kijken wanneer *alle* knopen in de giant component zitten), vinden we dat deze transitie ligt op $\langle k\rangle = 1$.

3.9.3 Een netwerk wordt volledig verbonden bij $\langle k\rangle \approx \ln N$

De kant dat een willekeurige knoop i níét in de giant component ligt, wordt gegeven door $(1-p)^{N_g}$. Omdat we al weten dat $N_G \approx N$ kunnen we dit afschatten met $(1-p)^N$. Het verwachte aantal van dit soort knopen is $I = N(1-p)^N = N(1-\frac{Np}{N})^N$. We weten dat $(1-\frac{x}{n})^n = e^{-x}$ voor $n \to \infty$, en dus is het aantal knopen niet in de giant component ongeveer gelijk aan $I \approx Ne^{-Np}$. We willen nu weten wanneer er nog maar één knoop niet in de giant component ligt (het transitiepunt), en hieruit lossen we makkelijk op

$$Ne^{-Np} = 1 \implies e^{-Np} = \frac{1}{N} \implies -Np = -\ln N \implies p = \frac{\ln N}{N} \implies \langle k\rangle \approx \ln N,$$

want $p = \langle k\rangle(N-1)$.