Extramaterial och matlab för Sannolikhet och statistik 2015

Johan Jonasson [*†‡]

Mars 2015

1 Simulering, transformering och några nya fördelningar

I Matlabs Statistics Toolbox finns färdiga kommandon för att generera stokastiska variabler enligt de flesta sannolikhetsfördelingar vi stött på. Därför kan de övningar som nu följer upplevas som omotiverade, men kunskapen om hur man transformerar slumptal av en viss fördelning till en annan, önskad, fördelning är inte bara av praktiskt utan också av teoretiskt intresse.

Antag att F_1 och F_2 är två fördelningsfunktioner. För enkelhets skull antar vi till en början att F_1 och F_2 är strängt växande, vilket betyder att de är inverterbara. Antag nu att X är en stokastisk variabel som är fördelad enligt F_1. Hur kan vi transformera X så att transformen får F_2 som fördelning? Svaret ges av

Proposition 1.1 *Om $X \sim F_1$ gäller att $Y = F_2^{-1}(F_1(X)) \sim F_2$.*

Att detta är sant ser vi av att

$$\begin{aligned}\mathbb{P}(Y \le y) &= \mathbb{P}\left(F_1(X) \le F_2(y)\right) = \mathbb{P}\left(X \le F_1^{-1}(F_2(y))\right)\\ &= F_1(F_1^{-1}(F_2(y))) = F_2(y).\end{aligned}$$

Ett specialfall av detta är att om $X \sim$ likf$[0,1]$, så är $F^{-1}(X) \sim F$. Detta är intressant eftersom datorgenerering av slumptal brukar utgå från likformig fördelning. I Matlab genererar man ett likf$[0,1]$-fördelat slumptal genom

[*]Chalmers University of Technology
[†]Göteborg University
[‡]jonasson@chalmers.se

```
>unifrnd(0,1)
```

Prova nu att generera en vektor `x` med 1000 sådana slumptal och åskådliggör dem på olika sätt, genom att plotta ett histogram och plotta den *empiriska fördelningsfunktionen*[1] Här kommer kommandona `hist` och `cdfplot` till pass.

Prova sedan att simulera enligt exponentialfördelning och normalfördelning för några olika parametrar, genom att transformera `x` enligt ovan. Normalfördelningens fördelningsfunktion är svår att invertera explicit, så till sin hjälp kan man ta kommandot `norminv`. **Leta nu också upp** kommandona som genererar direkt enligt dessa fördelningar och jämför, t.ex. genom att plotta de empiriska fördelningsfunktionerna i samma figur, eller de i storleksordning sorterade datavektorerna i samma figur. (Sorterar vektorer gör man med kommandot `sort`.)

Obs. Matlab parametriserar exponentialfördelningen med väntevärdet, dvs $1/\lambda$.

När F_1 och/eller F_2 inte är inverterbara, t.ex. om de svarar mot diskreta fördelningar, blir transformering aningen svårare, men bara marginellt eftersom man då istället kan använda sig av *generaliserade inverser*. Den generaliserade inversen till en fördelningsfunktion F ges av

$$\overline{F}(y) = \inf\{x : F(x) \geq y\}.$$

Eftersom F är högerkontinuerlig gäller att $F(\overline{F}(y)) = y$ för alla y, så propositionen ovan går igenom rakt av med inverser ersatta av generaliserade inverser.

Eftersom diskreta stokastiska variabler har fördelningsfunktioner på trappstegsform, är deras generaliserade inverser oftast knöliga att uttrycka. Som tur är har Matlab funktioner för generaliserade inverser till de vanligaste diskreta fördelningsfunktionerna. **Din uppgift** blir nu att finna dessa för binomialfördelning, geometrisk fördelning och Poissonfördelning och att transformera om `x` till dessa. **Prova** några olika parametervärden. **Finn också** Matlabfunktionerna för att direkt generera stokastiska variabler med dessa fördelningar och jämför med vad du fick m.hj.a. de generaliserade inverserna.

Nu är det dags att lära några nya kontinuerliga fördelningar: gamma, lognormal, χ^2, Weibull och Gumbel.

[1] Om $x_1, x_2, \ldots, x_n$ är ett datamaterial, t.ex. ett stickprov på någon stokastisk variabel, ges den empiriska fördelningsfunktionen till datamaterialet av fördelningsfunktionen till den stokastiska variabel Z för vilken $\mathbb{P}(Z = x_k) = 1/n$ för alla $k = 1, \ldots, n$.

Definition 1.2 *En stokastisk variabel X sägs vara gammafördelad med parametrar $\alpha > 0$ och $\lambda > 0$, $X \sim \Gamma(\alpha, \lambda)$, om den har täthetsfunktionen*

$$f(x) = \frac{1}{C}\lambda^{\alpha} x^{\alpha-1} e^{-\lambda x},\ x \geq 0$$

där C är den normaliserande konstanten $\Gamma(\alpha) := \int_0^{\infty} e^{-t} t^{\alpha-1} dt$

Om α är ett positivt heltal är det lätt att beräkna att $\Gamma(\alpha) = (\alpha - 1)!$. Notera att med $\alpha = 1$ får man tätheten för en exponentialfördelad stokastisk variabel, dvs

$$X \sim \Gamma(1, \lambda) \Leftrightarrow X \sim \exp(\lambda).$$

Faktum är att om $X_1, X_2, \ldots, X_n$ är oberoende och $\exp(\lambda)$-fördelade, gäller att

$$\sum_{k=1}^{n} X_k \sim \Gamma(n, \lambda). \tag{1}$$

Ta nu som uppgift att bevisa (1) m.hj.a. induktion. En omedelbar konsekvens av (1) är att tiden till den n:te impulsen i en Poissonprocess med intensitet λ är $\Gamma(n, \lambda)$-fördelad. Eftersom väntevärde och varians av en $\exp(\lambda)$-fördelad stokastisk variabel är $1/\lambda$ respektive $1/\lambda^2$, är motsvarande för en $\Gamma(n, \lambda)$-variabel n/λ respektive n/λ^2 och det är inte svårt att tro på att, allmänt,

$$X \sim \Gamma(\alpha, \lambda) \Rightarrow \mathbb{E}[X] = \frac{\alpha}{\lambda},\ \mathbb{V}\mathrm{ar}[X] = \frac{\alpha}{\lambda^2}.$$

Notera också att centrala gränsvärdessatsen ger att då n är stort, är $\Gamma(n, \lambda)$ mycket nära $\mathrm{N}(n/\lambda, (\sqrt{n}/\lambda)^2)$.

Generera nu $\Gamma(4, 1)$-fördelade slumptal på tre olika sätt: transformera `x` med `gaminv`, addera fyra $\exp(1)$-slumptal och genom att använda `gamrnd`. **Försök** också med simulering approximera sannolikheten att den tredje impulsen i en Poissonprocess med intensitet 2 kommer efter tid 2.4. Kolla också att du kan generera $\Gamma(\alpha, 1)$ även då α inte är ett heltal, t.ex. $\Gamma(2.47, 1)$.

Tack vare den centrala gränsvärdessatsen är det rimligt att anta att många naturligt förekommande storheter är normalfördelade. I vissa situationer är det dock uppenbart orimligt att göra ett sådant antagande. Framförallt sker detta i vissa, men långtifrån alla, situationer då man vet att den studerade storheten är positiv. Eftersom alla normalfördelningar har sitt stöd på hela den reella linjen,

kan inte en sådan storhet vara *exakt* normalfördelad. Ta till exempel den längd en son till en mor med given längd kommer att få i vuxen ålder. Om mamman är, säg, 170 cm kanske sonens förväntade längd är 184 cm med en standardavvikelse på 6 cm. Eftersom sonens längd naturligtvis är positiv, kan ett normalfördelningsantagande alltså inte vara exakt rätt. I detta fall ligger dock nollan mer än 28 standardavvikelser under väntevärdet, så att även om vi antar normalfördelning, kommer felet orsakat av risken att den antagna storheten blir negativ att bli ohyggligt litet. I det här fallet är det alltså (av det skälet) ingen fara att modellera sonens längd med en normalfördelning även om vi vet att det inte kan vara exakt rätt.

Titta nu istället på följande: en patient tar 75 mg av en viss medicin och man mäter sedan blodkoncentrationen X av det verksamma ämnet 2 timmar efteråt. Det är väl känt att upptaget av mediciner kan variera så mycket mellan olika patienter att man mycket väl kan ha t.ex. $\mathbb{E}[X] = 120$ (ng/ml), samtidigt som standardavvikelsen är 80 och nollan är bara 1.5 standardavvikelser från väntevärdet. Då blir felet med ett normalfördelningsantagandet alltför stort. I sådana situationer brukar man istället anta att *logaritmen* av X är normalfördelad, dvs att X går att skriva som $X = e^Y$ där Y är normalfördelad.

Definition 1.3 *Låt $Y \sim N(\mu, \sigma^2)$ och $X = e^Y$. Då säger man att X är lognormalfördelad med parametrar μ och σ^2, $X \sim logN(\mu, \sigma^2)$.*

Observera att det *inte* är sant att $X \sim \text{logN}(\mu, \sigma^2) \Rightarrow \mathbb{E}[X] = e^\mu$; faktum är att $\mathbb{E}[X] > e^\mu$ (om inte $\sigma^2 = 0$). Om $\sigma << |\mu|$ är skillnaden dock mycket liten. Man kan fråga sig om man inte alltid borde logaritmera positiva storheter innan man antar normalfördelning, som t.ex. sonens längd ovan. Svaret är: visst, det kan man absolut göra, men om $\sigma << \mu$ blir skillnaden så liten att det knappast är värt besväret! (Noga taget säger detta alltså att ibland är det rimligt att anta att både X och $\log X$ är normalfördelade. Är inte detta en motsägelse? Jo, båda kan inte vara *exakt* normalfördelade, men i praktiken är, om $\sigma << |\mu|$, båda väldigt nära normalfördelade.

Uppgift: Finn Matlabfunktionen som genererar lognormalfördelade slumptal direkt och kontrollera att du får samma resultat som om du genererar normalfördelade Y:n och sedan bildar e^Y. Om $X \sim \text{logN}(0, 1)$, vad ser det ut som att $\mathbb{E}[X]$ kan vara? Visst verkar det bli större än e^0?

En annan fördelning som uppstår ur normalfördelningen är χ^2-fördelningen.

Definition 1.4 *Låt $Z_1, Z_2, \ldots, Z_n$ vara oberoende och N$(0,1)$-fördelade och bilda*

$$X = \sum_{k=1}^{n} Z_k^2.$$

Man säger att X är χ^2-fördelad med n frihetsgrader och skriver $X \sim \chi_n^2$.

Det följer omedelbart att om $X_1, \ldots, X_n$ är oberoende och N(μ, σ^2) så gäller att

$$\frac{\sum_{k=1}^{n}(X_k - \mu)^2}{\sigma^2} \sim \chi_n^2.$$

Eftersom $Z \sim \mathrm{N}(0,1) \Rightarrow \mathbb{E}[Z^2] = 1$, ser vi att $\mathbb{E}[X] = n$ då $X \sim \chi_n^2$. Variansen ges av $\mathbb{V}\mathrm{ar}[X] = 2n$ (kan du göra denna beräkning?).

Faktum är, kanske överraskande, att χ^2-fördelningen är ett specialfall av gammafördelningen:

$$X \sim \chi_n^2 \Leftrightarrow X \sim \Gamma(n/2, 1/2).$$

Speciellt gäller alltså att om Z_1, Z_2 är oberoende och standardnormalfördelade, så är $Z_1^2 + Z_2^2 \sim \exp(1/2)$. **Bevisa** detta. **Använd** sedan resultatet till att generera standardnormalfördelade slumptal m.hj.a. exponentialfördelningen. Tips: Se Z_1 och Z_2 som koordinater för en vektor (Z_1, Z_2) och se på vektorns polär koordinater R och θ. Vad du just bevisade säger $R^2 \sim \exp(1/2)$. Man kan också visa att R och θ är oberoende och att $\theta \sim \mathrm{likf}[0, 2\pi]$. Nu kan du simluera R och θ och uttrycka Z_1 i termer av dessa.

Ett annat mycket intressant faktum är

Proposition 1.5 *Om $X_1, \ldots, X_n$ är oberoende och N(μ, σ^2) och vi som vanligt låter*

$$s^2 = \frac{1}{n-1}\sum_{k=1}^{n}(X_k - \overline{X})^2$$

så gäller att

$$\frac{(n-1)s^2}{\sigma^2} \sim \chi_{n-1}^2.$$

Vi bevisar inte detta. **Simulera** för att övertyga dig själv. Tag t.ex. $n = 10$, upprepa 1000 ggr och jämför med 1000 observationer av 1000 observationer genererade av `chi2rnd`. Att beräkna stickprovsvariansen för en datavektor är enkelt: använd bara kommandot `var`.

2 Felintensiteter, fördelningsplottar och några fördelningar till

Låt f vara täthetsfunktionen till en kontinuerlig stokastisk variabel $X \geq 0$. Minns att en tolkning av f är att om $h > 0$ är ett mycket litet tal, gäller att $\mathbb{P}(X \in (x, x+h)) \approx hf(x))$. Låt $G(x) = 1 - F(x) = \mathbb{P}(X > x)$.

Det gäller att

$$\mathbb{P}(X \in (x, x+h)|X > x) = \frac{\mathbb{P}(X \in (x, x+h))}{G(x)} \approx h\frac{f(x)}{G(x)}.$$

Man kan alltså tolka kvoten till höger som en intensitet för att "dö i nästa ögonblick givet att man överlevt tills nu". Därför är det inte ologiskt att man kallar den för X:s *felintensitet* eller *dödsintensitet*.

Definition 2.1 *Felintensiteten för X ges av*

$$r(x) = \frac{f(x)}{G(x)}.$$

Notera att om man känner till r så kan man beräkna G (och därmed hela X:s fördelning), eftersom $r(x) = -G'(x)/G(x) = -(d/dx)\ln G(x)$, vilket medför att

$$G(x) = \exp\left(-\int_0^x r(t)dt\right).$$

Antag att X har konstant felintensitet, λ. Då får vi $G(x) = e^{-\lambda x}$, dvs $X \sim \exp(\lambda)$. Detta är ingen överraskning, eftersom vi känner exponentialfördelningen som en livslängdsfördelning för saker som inte åldras. Detta antagande är ofta rimligt om man talar om t.ex. elektriska komponenter, men naturligtvis orimligt om man t.ex. talar om livslängder för människor eller djur. Det visar sig också ofta orimligt om man har att göra med data som är extremvärden av något slag, t.ex. månadens högsta vattenstånd eller kraftigaste vind. Vi ska se exempel på båda och börjar med det sistnämnda. **Ladda ner** filen `atlantic.dat`, som innehåller äkta data i form 582 s.k. signifikanta våghöjder uppmätta i Atlanten. Den signifikanta våghöjden definieras som medelvärdet av den högsta tredjedelen av vågorna. Data registeras 14 ggr per månad. Laddar ner gör du till exempel genom att skriva `y=load('atlantic.dat')` och får då data i vektorn `y`. Använd dessa data till att **skatta felintensiteten** som funktion av signifikant våghöjd. Detta kan man t.ex. göra med följande två kommandorader

```
for i=1:100, G(i)=length(y(y>0.12*i))/582;, end

for i=1:99, r(i)=(G(i)-G(i+1))/(0.12*G(i));, end
```

Här har vi delat in intervallet $[0, 12]$ i hundra lika delar och först skattat $G(x) = 1 - F(x)$ och sedan $f(x)/G(x)$. Plotta detta och se hur det ser ut. Verkar felintensiteten vara växande eller avtagande eller varken eller?

Definition 2.2 *Den stokastiska variabeln $X \geq 0$ sägs ha en Weibullfördelning med formparameter k och skalparameter λ, $X \sim Wbl(k, \lambda)$ om dess fördelningsfunktion, F, ges av*

$$F(x) = 1 - e^{-(\lambda x)^k}, x \geq 0.$$

Genom att derivera ser vi att om $X \sim \text{Wbl}(k, \lambda)$, är

$$f(x) = k\lambda(\lambda x)^{k-1} \exp(-(\lambda x)^k).$$

och att $r(x) = k\lambda(\lambda x)^{k-1}$, dvs felintensiteten är polynomiellt beroende av tiden. **Prova nu** att testa dina skattade felintensiteter från Atlantdata mot Weibullfördelingen för de parametrar som passar bäst till data. Använd funktionen `wblfit` för att finna bästa k och λ. Var observant på att Matlab parametriserar annorlunda än vi; vårt (k, λ) svarar mot $(1/\lambda, k)$ i Matlab.

Det du får bör inte se alltför illa ut, förutom att den skattade felintensiteten blir så vild att det är väldigt svårt att säga hur bra anpassningen är. Detta är ganska typiskt för skattade felintensiteter, så i själva verket är en jämförelse av felintensiteter inte ett bra sätt att se om ett stickprov kommer från en viss fördelning.

Ett bättre sätt är att jämföra den empiriska fördelningsfunktionen för data mot fördelningsfuntionen för den referensfördelning man vill jämföra (i detta fall Weibull) med med de bästa parametrarna. Detta kan man t.ex. göra genom att generera ett stickprov från referensfördelningen och jämföra med data, så som du redan gjort ovan vid ett flertal tillfällen. **Gör detta** för Atlantdata och plotta i samma figur med `cdfplot`. Betydligt tydligare, eller hur?

Ännu bättre är dock att jämföra direkt med fördelningsunktionen för referensfördelningen. Ett speciellt sätt att göra det är att göra **fördelningsplot** av data. Detta betyder att man plottar den empiriska fördelningsfunktionen för data i ett diagram, där axlarna skalas så att om data väl överensstämmer med referensfördelingen, så kommer de att hamna nära en rät linje i figuren. Sådana plotfunktioner finns i Matlab för att jämföra med en några olika fördelningar. I detta fall **ska det alltså göras** en Weibullplot och kommandot är `wblplot`. Du bör se att överensstämmelsen är hyfsad, men att det ändå blir tydligt fel i svansarna.

Definition 2.3 *Man säger att X är Gumbelfördelad med skalparameter a och lägesparameter b, $X \sim$ Gumb(a, b), om*

$$F(x) = \exp(-e^{-(x-b)/a}), x \in \mathbb{R}.$$

Prova att testa Atlantdata mot Gumbelfördelningen på samma sätt som för Weibull, genom att dels jämföra med simulerade data från Gumbelfördelningen, dels gäre en Gumbelplot. Eftersom Matlab inte exakt har färdiga kommandon för detta kan du ladda ner m-filerna `gumbfit`, `gumbcdf` och `gumbplot` från kurshemsidan. Förhoppningsvis finner du att Atlantdata passar bättre med Gumbel än med Weibull.

En mycket viktig egenskap hos Gumbelfördelningen är att den är *max-stabil*, dvs om X_1 och X_2 är oberoende och Gumbel med samma skalparameter, så är även $\max(X_1, X_2)$ Gumbel med samma skalparameter (men med en annan lägesparameter, vilken då?). **Bevisa detta**. Som specialfall får vi att om $X_1, X_2, \ldots, X_n$ är oberoende och likafördelade Gumbel, så är $\max(X_1, \ldots, X_n)$ också Gumbel med samma skalparameter. Detta är ett skäl till att Gumbelfördelningen passar bra som modell till somliga extremvärdesdata.

Låt oss nu titta på felintensiteter för ett helt annat datamaterial, nämligen ett med livslängder hos människor. **Ladda ner** data från filen `norway.mat` genom att bara skriva `load('norway.mat')`. Du får data i matrisen `norway`. I den andra kolonnen i matrisen `norway` finns skattade överlevnadssannolikheter för norska kvinnor och i den tredje kolonnen finns motsvarande siffror för norska män. Data ska tolkas så att om talet i element n är x gäller att av 100000 hypotetiska kvinnor (män) födda år 2000 överlever x av dem sin $n-1$:a födelsedag. Använd dessa data till att skatta dödsintensiteten för norska män/kvinnor som funktion av ålder. Plotta båda i samma figur och se om du ser någon skillnad.

Kolla nu om data passar med normalfördelning, Weibull eller Gumbel. Inget vidare, va? Faktum är att data stämmer mycket bättre överens med en fördelning som har dödsintensitet

$$r(t) = a + be^{t/c}.$$

I det aktuella fallet är de värden på parametrarna a, b och c som passar bäst $a = 9 \cdot 10^{-4}$, $c = 10.3$ och $b = 4.4 \cdot 10^{-5}$ för män och $b = 3.3 \cdot 10^{-5}$ för kvinnor. (Denna modell används av livförsäkringsbolag.) Vad är, i denna modell med dessa a, b, c, den betingade sannolikheten att en norsk man född år 200), blir minst 80 år givet att han blir minst 30 år?

3 Poissonprocessen

Poissonprocessen är en process av *impulser* som inträffar på ett sådant sätt att tiderna mellan två impulser är oberoende och exponentialfördelade med samma parameter λ. Parametern λ kalla då för Poissonprocessens *intensitet*. Man inser att det är lätt att simulera en Poissonprocess med en given intensitet λ; låt bara $T_1, T_2, \ldots$ vara oberoende och exp(λ)-fördelade och låt tidpunkten S_n för den n:te impulsen ges av $T_1 + \ldots + T_n$, $n = 1, 2, \ldots$. Skriv $X(s,t)$ for antalet impulser in tidsintervallet (s,t) och $X(t)$ för $X(0,t)$. Du har sett (eller kommer att se) att $X(s,t)$ är Poissonfördelad med parameter $\lambda(t-s)$. **Testa detta** genom att, enligt ovan, simulera en Poissonprocess med intensitet 2 och se hur många impulser du får mellan tidpunkterna 2 och 5. Upprepa simuleringen, säg, 1000 ggr och registera $X(2,5)$ varje gång. Gör histogram och jämför med vad du får om du simulerar den väntade Poissonfördelningen. Man har god hjälp av att veta att kommandona för att simulera exponential- respektive Poissonfördelade data är `exprnd` respektive `poissrnd`.

Två andra centrala egenskaper hos Poissonprocessen är *sammanvägningsegenskapen* och *uttunningsegenskapen.* Sammanvägningsegenskapen säger att om $X(t)$ och $Y(t)$ är antal impulser i två oberoende Poissonprocesser med intensitet λ_x respektive λ_y, så ä $X(t) + Y(t)$ antalet impulser i en Poissonprocess med intensitet $\lambda_x + \lambda_y$. (M.a.o. den sammanvägda processen är den som räknar som impulser de som sker vid tidpunkter då en impuls sker i antingen $X(t)$ eller i $Y(t)$.) **Prova detta** genom att simulera en sammanvägd Poissonprocess, registrera tidsavstånden mellan impulserna, plotta histogram och se om det tycks stämma med simulering av den väntade exponentialfördelningen. Ett bra sätt att kontrollera om en vektor `x` tycks innehålla exponentialfördelade data är, förutom att simulera exponentialfördelade data och jämföra med, att göra en exponentialfördelningsplot, genom att skriva `probplot('exponential',x)`. Om data hamnar längs en rät linje i ploten tyder detta på exponentialfördelning, medan avvikelser från detta tyder på annan fördelning.

Observera att sammanvägningsegenskapen faktiskt är ekvivalent med egenskapen hos exponentialfördelningen att om X och Y är oberoende, $X \sim \text{exp}(\lambda_x)$ och $Y \sim \text{exp}(\lambda_y)$ så är $\min(X,Y) \sim \text{exp}(\lambda_x + \lambda_y)$. (Varför?)

Uttunningsegenskapen säger att om $X(t)$ är en Poissonprocess med intensitiet λ och $Y(t)$ är den process som ges av att varje impuls i $X(t)$ räknas som impuls i $Y(t)$ med en given sannolikhet p, oberoende av andra impulser. **Testa även** detta på samma sätt.

Poissonprocessen används för alla typer av processer, där man bedömer att, för alla tidpunkter, varken tiden sedan senaste impuls eller tidpunkten i sig påverkar fördelningen av tiden till nästa impuls. Det kan vara trafikolyckor, stormar, jordbävningar, mål i en fotbollsmatch, grupper av människor du möter på din kvällspromenad, radioaktiva sönderfallsprocesser etc. Låt oss titta på ett exempel med riktiga data. **Ladda ner** filen `coal.dat`. Denna fil innehåller information om dödsolyckor i brittiska kolgruvor från 1851 till 1918 och utgörs av en matris med sex kolonner, där de olika kolonnerna står för följande data kring de skedda dödsolyckorna under mätperioden.

1. Dag i månad.
2. Månad.
3. År.
4. Dagnummer på aktuellt år.
5. Antal dagar sedan senaste olyckan.
6. Antal förolyckade.

Har olyckorna kommit som en Poissonprocess? Testa genom att se på tiderna mellan olyckor och jämföra med exponentialfördelning. Det bör se i stort sett bra ut, men om man tittar närmare bör man också se att det verkar vara lite för många korta och långa tider mellan olyckor jämfört med vad man får med exponentialfördelning. Vi har råkat ut för att vi observerat en *tidsinhomogen* Poissonprocess, dvs intensiteten har ändrat sig med tiden (dvs vi har en process där, vid en given tidpunkt, tiden sedan senaste olyckan inte påverkar tiden till nästa olycka, men att den innevarande tidpunkten i sig har betydelse.) Vi ska inte gå närmare in på teorin bakom dessa, mer än att konstatera att situationen är mycket vanlig och den tidshomogena varianten, dvs den vi studerat hittills, är ett bekvämt specialfall. Om vi ser på de nämnda processerna ovan, är det i jordbävningsfallet naturligt med att tro på tidshomogenitet (om vi inte tittar över geologiska tidsspann), medan det i de andra fallen är mer eller mindre realistiskt beroende på tids- och/eller rumsspann. (Exempelvis har stormintensiteten en årstidsvariation, det görs mer mål senare i fotbollsmatcher, man möter eventuellt mer folk tidigare under promenaden, etc, men tidsvariationerna blir inte betydelsefulla förrän man studerar tillrčckligt långa tidsspann.)

Om man tittar närmare på data i fallet med olyckorna i kolgruvorna, kan man se ett trendbrott kring olycka nr 127. **Prova att** studera data före respektive detta, var för sig. Ser det ut som att det är skillnad i intensitet? (Runt 1880 infördes nya säkerhetsrutiner för att minska olycksrisken. Frågan är om detta fick effekt.)

4 Tester av normalfördelade data, p-värden och styrkor

En ”standardsituation” när man gör konfidensinterval och tester är att man arbetar under antagandet att data är normalfördelade. Vi har sett detta både i enstickprovsfall och tvåstickprovsfall. I enstickprovsfallet antar man att man har att stickprov $X_1, \ldots, X_n$ med $X_k \sim \mathrm{N}(\mu, \sigma)$, där μ är okänd och σ kan vara känd eller okänd. Låt oss här anta det vanligaste fallet, nämligen att σ är okänd. Vi är intresserade av att göra konfidensintervall för μ eller testa nollhypotesen $H_0 : \mu = \mu_0$ mot alternativhypotesen $H_A : \mu \neq \mu_0$ eller $H_A : \mu > \mu_0$ eller $H_A : \mu < \mu_0$. Kom ihåg korrespondensen mellan konfidensintervall och test; H_0 förkastas på signifikansnivå α om och endast om motsvarande konfidensintervall med konfidensgrad $1 - \alpha$ inte innehåller μ_0. Här svarar ett symmetriskt konfidensintervall

$$\mu = \overline{X} \pm F^{-1}_{t_{n-1}}(1 - \alpha/2)\frac{s}{\sqrt{n}}$$

mot det tvåsidiga testet som har $H_A : \mu \neq \mu_0$ som alternativhypotes, medan alternativet $H_A : \mu > \mu_0$ svarar mot det nedåt begränsade konfidensintervallet

$$\mu \geq \overline{X} - F^{-1}_{t_{n-1}}(1 - \alpha)\frac{s}{\sqrt{n}}.$$

Även här ska vi jobba med riktiga data. **Ladda ner** filen `birth.dat`, genom i Matlabföntret gå in under ”Import data” och där välja ”birth” och sedan ladda ner. Data är lagrade i en 747×26-matris och innehåller en mängd uppgifter om nyfödda barn till förstföderskor på fyra mödravårdscentraler i Malmö åren 1991 och 1992. Vilka uppgifter filen innehåller finns beskrivet i filen ”birth.txt” (med undantaget att uppgiften ”.” för ”data saknas” ersatts av ”99” för att Matlab inte ska protestera).

Vi ska mest intressera oss för de nyfödda barnens födelsevikt, så se till att skapa en vektor `fv=birth(:,3)` med dessa. Vi vill anta att dessa är normalfördelade. Är det ett rimligt modellantagande? **Gör en normalfördelningsplot** med `normplot(fv)`. Det bör se ganska bra ut men inte helt perfekt. Framförallt bör det synas att det finns fler barn än väntat (utifrån normalfördelningsantagandet)

som har riktigt låg födelsevikt. Det verkar hursomhelst vara hyfsat bra med ett normalfördelningsantagande, så låt oss jobba vidare under det antagandet.

Man brukar som tumregel säga att väntevärdet av vikten hos ett på måfå valt nyfött svenskt barn till en förstföderska är 3500 g (med en standardavvikelse på ca 500 g). Stämmer detta med vårt datamaterial? **Testa**, på 5% signifikansnivå nollhypotesen $H_0 : \mu = 3500$ mot $\mu_A : \mu \neq 3500$ och ge ett 95% konfidensintervall för μ.

Det finns olika sätt att göra detta i Matlab. Ett sätt är att använda kommandot `normfit`. Det bekvämaste är dock att använda funktionen `ttest`; den kan nämligen besvara alla frågorna på en gång! **Skriv**

$$[\mathtt{h}, \mathtt{p}, \mathtt{ki}] = \mathtt{ttest}(\mathtt{fv}, 3500, \mathtt{a})$$

och Matlab svarar med att ge

- `h`: förkastelseindikator, dvs $h = 1$ om H_0 ska förkastas till förmån för alternativet $\mu \neq 3500$ på signifikansnivå a och $h = 0$ annars,
- `p`: testets p-värde, dvs den lägsta signifikansnivå på vilken H_0 hade kunnat förkastas med dessa data.
- `ki`: undre och över grnser för det symmetriska konfidensintervallet för μ med konfidensgrad $1 - a$.

Du bör komma fram till ett p-värde på ca $2.2 \cdot 10^{-6}$, så testet är kraftfullt signifikant, och ett 95% konfidensintervall ges av $\mu = 3400 \pm 41$. Uppgiften $\mu = 3500$ verkar alltså inte stämma utan medelvikten tycks snarare ligga kring 3400 g.

OK, men nu är det ju så att alla barn tagits med i denna studie, även de som är tidigt födda. Man brukar klassa den som föds efter mindre än 266 graviditetsdagar som tidigt född. **Prova nu** att testa bara på de som inte är tidigt födda. Uppgifter om graviditetens längd finns i kolonn 1 i `birth`. Du kommer att se att resultatet blir annorlunda. Om du också gör en normalfördelningsplot, kommer du även att se att normalfördelningsantagandet passar mycket bättre för dessa data. Kan du förklara hur detta kan komma sig?

Låt oss nu fokusera på tvåstickprovsfallet. Här har man två oberoende stickprov $X_1, \ldots, X_n$ och $Y_1, \ldots, Y_m$ där $X_k \sim \mathrm{N}(\mu_x, \sigma_x^2)$ och $Y_j \sim \mathrm{N}(\mu_y, \sigma_y^2)$ och är oftast intresserad av att testa $H_0 : \mu_x = \mu_y$ mot $H_A : \mu_x \neq \mu_y$. För att klara detta

analytiskt brukar man anta att $\sigma_x^2 = \sigma_y^2 = \sigma^2$ och utnyttja att

$$\frac{\overline{X} - \overline{Y} - (\mu_x - \mu_y)}{s_P\sqrt{1/n + 1/m}} \sim t_{n+m-2},$$

där s_P^2 är den poolade stickprovsvariansen given av

$$s_P^2 = \frac{(n-1)s_x^2 + (m-1)s_y^2}{n+m-2}.$$

Utifrån detta skapas t.ex. det symmetriska konfidensintervallet

$$\mu_x - \mu_y = \overline{X} - \overline{Y} \pm F_{t_{n+m-2}}^{-1}(1-\alpha/2)s_P\sqrt{\frac{1}{n} + \frac{1}{m}}$$

med konfidensgrad $1-\alpha$ och H_0 testas på motsvarande sätt.

I Matlab finns funktionen `ttest2` som, precis som `ttest`, kan besvara alla frågorna samtidigt. Använd Matlabs hjälpfunktion för att se hur kommandot fungerar. (Som du ser kan kommandot även hantera situationer där $\sigma_x^2 \neq \sigma_y^2$ (numeriskt).) **Testa nu** om det tycks föreligga någon skillnad i förväntad födelsevikt mellan flickor och pojkar. Gör det samma för rökare mot ickerökare och för mödrar som sammanbor med barnets far och de som inte gör det och gärna lite fler saker om du har tid och lust. (Uppgifter om barnets kön, moderns rökvanor och ev. smmanboende med fadern finns i kolonnerna 2, 20 respektive 18.)

Anmärkning: Kom ihåg multipelinferensproblemet: ”i alla datamaterial finns det alltid något som är signifikant”. Detta hade vi verkligen behövt tänka på om vi skulle testa så här friskt i en ”skarp” situation. Låt oss därför betrakta vår övning som en hypotesgenererande pilotstudie.

Återvänd nu till det allmänna enstickprovsfallet och betrakta styrkan i testet $H_0 : \mu = \mu_0$ mot $H_A : \mu \neq \mu_0$. Testet baseras på att om H_0 är sann gäller

$$\frac{\overline{X} - \mu_0}{s/\sqrt{n}} \sim t_{n-1}$$

och förkastar H_0 på signifikansnivå α om

$$\frac{|\overline{X} - \mu_0|}{s/\sqrt{n}} > F_{t_{n-1}}^{-1}(1-\alpha/2).$$

Styrkan för μ_1, $g(\mu_1)$, är sannolikheten att testet förkastar H_0 om μ_1 är det sanna värdet på μ. Det uppstår två svårigheter, dels att för att veta fördelningen hos teststatistikan $T = (\overline{X} - \mu_0)/(s/\sqrt{n})$ då $\mu = \mu_1$, måste vi känna σ^2, dels att då $\mu = \mu_1$ är T inte t-fördelad utan har en s.k. *ickecentral t-fördelning*. Den ickecentrala t-fördelningen finns naturligtvis tabellerad och installerad i all god statistisk programvara. Ett problem är dock att en parameter i denna fördelning är σ^2, den okända variansen. Eftersom styrkeberäkningar används främst för att dimensionera studier så att man upptäcker avvikelser av given storlek med (approximativ) önskad sannolikhet, måste denna analys göras innan man har tillgång till måtdata och utifrån dessa skatta σ^2. Man är hänvisad till att göra en kvalificerad gissning av vad σ^2 är. Om man har tur finns det tidigare undersökningar av liknande slag att hämta erfarenhet från, annars får man gissa påannan grund.

Anmärkning. Som man förstår är det ofta svårt att göra riktigt objektiva styrkeberäkningar. Icke desto mindre är de ytterst viktiga. Betrakta ett läkemedelsbolag som vill prova en ny medicin och man vill se om denna har en effekt som är bättre än placebo eller eventuella existerande lk̈emedel. Vi tänker oss att vi står inför den skarpa fas 3-studien där medicinens effekt slutligen ska testas på mn̈iskor i ett fullskaligt försök. Man identifierar en viss minsta effekt som man tycker är värd att upptäcka, den s.k. minsta kliniskt signifikanta effekten. Man vill dimensionera försöket så att man har, säg, 90% chans att upptäcka en sådan effekt (om den är den sanna effekten; chansen blir förstås större om den sanna effekten är större n̈ så). Eftersom studier av detta slag är oerhört dyra, vill man absolut inte göra studien större än nödvändigt. Å andra sidan vill man naturligtvis heller inte göra studien så liten att man löper en stor risk att missa att upptäcka en effekt som egentligen är signifikant; det blir ju att kasta pengarna i sjön! Man är alltså i stort behov av en god styrkeberäkning (och en god beräkning av kostnad av att misslyckas på olika sätt som ska vägas mot vad man beräknar att man kan vinna på medicinen om den kommer ut på marknaden).

Titta nu som exempel på studien av nyfödda barn. (Vi får tn̈ka oss att vi ännu inte har tillgång till några mätdata.) Hur många barn skulle behövt ingå i studien för att sannolikheten att förkasta $H_0 : \mu = 3500$ med minst 75% chans om vi testar på signifikansnivån 5% och det sanna värdet på μ är 3450? Som gissning av σ^2 använder vi tumregeln $\sigma = 500$. Komandot `sampsizepwr` hjälper till. Kommandot kan också användas till att bestämma styrkan för givet n (och μ_1, μ_0, σ_2).

5 Linjär regression

Kom ihåg att vi har parade data (x_k, Y_k), $k = 1, \ldots, n$, där x_k:na är kända storheter och Y_k:na antas bero linärt av x_k:na, med ett normalfördelat slumpfel, oberoende för olika k, dvs $Y_1, \ldots, Y_n$ är oberoende och det finns konstanter a och b sådana att

$$Y_k \sim \mathrm{N}(a + bx_k, \sigma^2).$$

Detta kan också skrivas som $Y_k = a + bx_k + \epsilon_k$ där ϵ_k:na är oberoende och $\mathrm{N}(0, \sigma^2)$-fördelade slumpfel.

Man vill utifrån data skatta de okända parametrarna a och b (och σ^2). Om man använder sig av Maximum-Likelihood-metoden, visar det sig man finner punkskattningarna $\hat{a}$ och $\hat{b}$ av a och b genom att minimera

$$\sum_{k=1}^{n} (Y_k - (a + bx_k))^2,$$

dvs ML-metoden sammanfaller med minsta-kvadrat-metoden för att att approximera det överbestämbda ekvationssystemet $Y_k = a + bx_k$, $k = 1, \ldots, n$ (med a och b som variabler och x_k:na och Y_k:na sedda som kända koefficienter). Man får också att $\hat{a}$ och $\hat{b}$ har normalfördelningar, som beror av σ^2, vilket i sin tur sedan ger att man kan göra konfidensintervall för a och b m.hj.a. t-fördelningen (eller normalfördelningen om σ^2 är känd).

Det enklaste sättet att i Matlab få värdena på ML-skattningarna $\hat{a}$ och $\hat{b}$ är att skriva `polyfit(x,Y,1)`, där `x` och `Y` förstås är vektorerna med x- respektive Y-värdena. För en visuell framställning, skriv `scatter(x,Y)` eller `plot(x,Y,'.')` (vilket jag tycker är snyggare) och ge kommandot `lsline` för att plotta regressionslinjen genom datapunktsvärmen.

För att få konfidensintervall, får man krångla lite mer. Kommandot att använda är `regress`, vilket baserar sig på den *generella linjära modellen* (GLM), som vi inte sett i denna kurs. Linjär regression kan ses som ett specialfall av GLM, genom att skriva $Y_k = a + bx_k + \epsilon_k$, $k = 1, \ldots, n$ på matrisform som

$$Y = C\beta + \epsilon$$

där $Y = [Y_1 \ldots Y_n]^T$, $\beta = [a\ b]^T$, $\epsilon = [\epsilon_1 \ldots \epsilon_n]^T$ och C är $n \times 2$-matrisen som har kolonnerna $[1 \ldots 1]^T$ och $x = [x_1 \ldots x_n]^T$. Kommandot `regress` tar Y och C som indata och du skapar C genom att skriva `C=[ones(n,1) x]`. **Använd nu** detta till att göra en linjär regression av födelseviktsdata mot graviditetslängd,

genom att låta x vara vektorn av graviditetslängder (dvs antalet graviditetsdagar före födsel, i kolonn 1 av `birth`) och Y födelsevikterna. Vad blir konfidensintervallet för b med konfidensgrad 99%? Du bör få $\hat{b} \approx 28$. Hur tolkar du detta värde?

6 Variansanalys

En generalisering av tvåstickprovsfallet är en situation där man har fler än två stickprov från, möjligen, olika normalfördelningar. Situationen är följande: Vi har k stycken oberoende stickprov

$$X_{11}, X_{12}, \ldots, X_{1n_1},$$
$$X_{21}, X_{22}, \ldots, X_{2n_2},$$
$$\vdots$$
$$X_{k1}, X_{k2}, \ldots, X_{kn_k}$$

där stickprov nr i, dvs $X_{i1}, \ldots, X_{in_i}$ är tagen från en $\mathrm{N}(\mu_i, \sigma^2)$-fördelning. Vi antar alltså att variansen är densamma för alla stickprov. Vi vill testa $H_0 : \mu_1 = \mu_2 = \ldots = \mu_k$ mot alternativhypotesen att inte alla väntevärden är desamma.

Exempel. I ett flervåningshus vill man kontrollera om det finns en våningseffekt på radonhalten i lägenheterna. Detta skulle kunna tyda på en källa till radonet är berggrunden under huset. På de olika våningarna placerar man ut ett antal mätdosor och vill med mätdata pröva om en våningseffekt finns, dvs man vill testa nollhypotesen $\mu_1 = \ldots = \mu_k$ mot alternativhypotesen att ej alla μ_i är lika. Här är förstås μ_i sann genomsnittlig radonhalt på våning i.

Naturligtvis skulle man kunna använda tvåstickprovstest för alla par av stickprov, men då stöter man genast på de problem det innebär att testa många hypoteser på samma datamaterial. I den situation vi har här, där man vill testa mot den ganska svaga alternativhypostesen som bara säger att *något* μ_i skiljer sig från de andra, kan man utnyttja data på ett betydligt effektivare sätt. Den teststatistika man brukar använda för detta test bygger på idén att om variationen hos data *mellan* stickproven är stor i förhållande till variationen *inom* stickproven, tyder detta på att de inte kommer från samma fördelning, dvs att H_0 inte är sann.

Låt $n = \sum_{i=1}^{k} n_i$, det totala antalet data. Skriv

$$\overline{X}_{i\cdot} = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij}$$

för medelvärdet av stickprov i och $\overline{X}$ för medelvärdet av alla data, dvs

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{k}\sum_{j=1}^{n_i} X_{ij} = \frac{1}{n}\sum_{i=1}^{k} n_i \overline{X}_{i.}$$

Skriv också

$$s_i^2 = \frac{1}{n_i - 1}\sum_{j=1}^{n_i}(X_{ij} - \overline{X}_{i.})^2$$

för stickprovsvariansen i stickprov i. Enligt Proposition 1.5 gäller att $(n_i - 1)s_i^2/\sigma^2 \sim \chi^2_{n_i-1}$ för alla i. Genom att summera följer att

$$\sum_{i=1}^{k} \frac{(n_i - 1)s_i^2}{\sigma_2} \sim \chi^2_{n-k}.$$

Genom att skriva

$$s_W^2 = \frac{1}{n-k}\sum_{i=1}^{k}(n_i - 1)s_i^2 = \frac{1}{n-k}\sum_{i=1}^{k}\sum_{j=1}^{n_i}(X_{ij} - \overline{X}_{i.})^2$$

blir detta

$$\frac{(n-k)s_W^2}{\sigma^2} \sim \chi^2_{n-k}.$$

Här står indexet W för ”within”, vilket ska markera att vi tänker på s_W^2 som den poolade (dvs sammanvägda) stickprovsvariansen *inom* stickproven.

Vidare gäller $\overline{X}_{i.} \sim \mathrm{N}(\mu_i, \sigma^2/n_i)$. För att förenkla en aning, låt oss för en stund anta att alla stickprov är lika stora, dvs det finns ett m så att $n_i = m$ för alla i. Om nollhypotesen att alla μ_i är lika, skriv μ för deras gemensamma värde och observera att då är $\overline{X}_{1.}, \ldots, \overline{X}_{k.}$ ett stickprov på $\mathrm{N}(\mu, \sigma^2/m)$. Därför ger Proposition 1.5 att

$$\frac{\sum_{i=1}^{k} m(\overline{X}_{i.} - \overline{X})^2}{\sigma^2} \sim \chi^2_{k-1}.$$

Genom att skriva

$$s_B^2 = \frac{1}{k-1}\sum_{i=1}^{k} m(\overline{X}_{i.} - \overline{X})^2$$

blir detta

$$\frac{(k-1)s_B^2}{\sigma_2} \sim \chi^2_{k-1}.$$

Indexet B står för "between", för att markera att vi tänker på s_B^2 som stickprovsvariansen *mellan* stickproven. Återgå nu till fallet med (möjligen) olika n_i:n. Man generaliserar då s_B^2 till

$$s_B^2 = \frac{1}{k-1}\sum_{i=1}^{k} n_i(\overline{X}_{i\cdot} - \overline{X})^2.$$

Det går att återigen visa (vilket vi dock inte gör här) att $(k-1)s_B^2/\sigma^2 \sim \chi^2_{k-1}$. Det går också att visa att s_W^2 och s_B^2 är oberoende. Det betyder att om H_0 är sann, har kvoten $T := s_B^2/s_W^2$ en s.k. *F-fördelning*:

Definition 6.1 *Låt Y_1 och Y_2 vara två oberoende stokastiska variabler sådana att $Y_1 \sim \chi^2_{m_1}$ och $Y_2 \sim \chi^2_{m_2}$. Då sägs kvoten*

$$\frac{Y_1/m_1}{Y_2/m_2}$$

vara F-fördelad med m_1 och m_2 frihetsgrader, skrivet

$$\frac{Y_1/m_1}{Y_2/m_2} \sim F_{m_1,m_2}.$$

Det följer direkt av denna definition att om H_0 är sann gäller

$$T = \frac{s_B^2}{s_W^2} \sim F_{k-1,n-k}.$$

Om man får ett stort värde på T talar detta för H_A snarare än H_0. Testet av H_0 : "alla μ_i lika" mot H_A: "ej alla μ_i lika" ges därmed av att förkasta H_0 till förmån för H_A på signinfikansnivå om

$$T \geq F^{-1}_{F_{k-1,n-k}}(1-\alpha).$$

Uppgift: Ett litet bussbolag med endast fem bussar, av samma märke, vill prova fyra olika däckstyper med avseende på slitstyrka och se om det spelar någon roll vilken däckstyp man väljer. Man genomför ett experiment där man sätter ett däck av varje typ på var och en de fem bussarna. När bussarna kört 2000 mil, mäter man slitaget på däcken (i millimeter). Detta ger ett stickprov $X_{i1}, \ldots, X_{i5}$ per däckstyp i, $i = 1, 2, 3, 4$. Mätdata blev:

Däckstyp 1: 9.1 13.4 15.6 11.0 17.1

Däckstyp 2: 20.3 20.3 24.6 18.2 19.8

Däckstyp 3: 20.8 28.3 23.7 21.4 25.1

Däckstyp 4: 11.8 16.0 16.2 14.1 15.8

Kan vi anta att dessa är fyra stickprov på normalfördelningar med samma varians? Gör en normalfördelningsplot för varje stickprov. Det ser inte alltför illa ut, så låt oss anta normalfördelning. Beräkna nu stickprovsvarienserna. De ser inte ut att skilja sig så mycket, så låt oss också anta lika varians. Vi ska dock vara medvetna om att stickproven är små, så de här beräkningarna ger inte särskilt mycket information och antagandena är klart osäkra. Eftersom de olika bussarna är av samma märke, verkar det också rimligt att anta att stickproven är oberoende (eftersom man det då verkar hyfsat troligt att variationen i slitage mellan däcken utgörs av skillnader hos däcken själva snarare än vilken buss de sitter på men även detta är ett ganska osäkert antagande).

Syftet med försöket tycks vara att testa $H_0 : \mu_1 = \ldots = \mu_4$ mot H_A: ”ej alla μ_i lika”. Vi har ett försök med *balanserad design*, dvs alla stickprov innehåller lika många observationer. För sådana fall fungerar Matlabs kommando `anova1` mycket smidigt; låt bara stickproven utgöra varsin kolonn i matrisen A och skriv `anova1(A)`. Matlab svarar med testets p-värde (och en figur och en tabell som man inte behöver bry sig om här.) Jämför gärna med p-värden för parvisa tvåstickprovs t-tester (utan att fundera alltför djupt över vad jämförelsen säger).

Anmärkning. På engelska heter variansanalys ”Analysis of Variance”, förkortat ”ANOVA”, därav namnet `anova1` på Matlabs kommando. Ettan står för ”1-vägs design”; det finns även 2-vägs design (och n-vägs design), en något annorlunda modell som vi inte går in på här.

Uppgift: I matrisen `birth` med födelseviktsdata finns bl.a. en uppgift om moderns ålder vid barnets födelse. Man har angivit med en etta om modern var 15-24 år, en tvåa om hon var 25-29 år och med en trea om hon var 30-44 år. Uppgifterna finns i den åttonde kolonnen. Dela nu födelseviksdata i tre grupper baserade på ålderskategori och genomför en variansanalys för att se om det finns en effekt av ålderskategori på födelsevikterna.

Du kommer att finna att det i detta fall rör sig om en obalanserad design. Matlab har inte något lika smidigt kommando för denna situation, så du får beräkna värdet på den F-fördelade teststatistikan och stoppa in detta i fördelningsfunktionen för rätt F-fördelning. Denna fördelningsfunktion kan Matlab ge; kommandot är `fcdf`.

7 Ickeparametriska metoder

I vissa situationer kan det vara så att man inte kan försvara något fördelningsantagande för data. Då finns det ändå vissa saker man kan göra, vilket vi ska se exempel på här.

Antag att $X_1, \ldots, X_n$ är ett stickprov på en stokastisk variabel, som vi antar är kontinuerlig, men i övrigt inte vet något om.

Definition 7.1 *Låt X vara en stokastisk variabel med fördelningsfunktion F. Då kallas talet m för en median till X (eller F) om $F(m) = 1/2$.*

Eftersom fördelningsfunktionen till en kontinuerlig stokastisk variabel är kontinuerlig, gäller att varje kontinuerlig stokastisk variabel har en median. Den är inte nödvändigtvis unik, men om F är strikt växande gäller även detta. Observera att medianen *inte* är samma sak som väntevärdet. Om X har *symmetrisk* täthetsfunktions kring m, dvs $f(m + x) = f(m - x)$ för alla x, och X har ett väntevärde, gäller att $m = \mathbb{E}[X]$, men alltså inte säkert annars. Se till exempel på fallet $X \sim \exp(1)$. Då är $\mathbb{E}[X] = 1$, men $P(X > x) = e^{-x} = 1/2$ då $x = \ln 2$, så $m = \ln 2$.

Antag för enkelhets skull att stickprovsvariablen X har en unik median (för att undvika onödig förvirring, men allt vi ska göra funkar även utan detta antagande). Medianen m är alltså ett tal sådant att $\mathbb{P}(X < m) = \mathbb{P}(X > m) = 1/2$. Fixera ett tal m_0. Låt N_+ vara antalet k sådana att $X_k > m_0$. *Om* $m = m_0$ blir $N_+ \sim \text{Bin}(n, 1/2)$ och bör hamna nära $n/2$. En alltför stor avvikelse från detta tyder på att $m \neq m_0$. Det är upplagt för ett test; välj c så att

$$F_{\text{Bin}(n,1/2)}(n/2 + c) \geq 1 - \alpha/2$$

och förkasta $H_0 : m = m_0$ till förmån för $H_A : m \neq m_0$ på signifikansnivå högst α om

$$|N_+ - n/2| > c.$$

(Att vi oftast inte kan få exakt signifikansnivå α beror på att binomialfördelningen är diskret.) Detta test kallas för ett *teckentest* av medianen. Matlabs kommando heter `signtest`. **Prova detta** på födelsedata och testa om medianen är 3400 g.

Att $N_+ \sim \text{Bin}(n, 1/2)$ kan naturligtvis också användas för att göra parameterfritt konfidensintervall för m.

Anmärkning. Vektorn `fv` med födelsevikter har medelvärde 3400 g och median 3430 g. Vi har ju antagit att data är normalfördelade och eftersom normalfördelningen är symmetrisk sammanfaller väntevärde och median och vi förväntar

oss att de skattade storheterna ska vara mycket nära varandra i datamaterialet. Detta stämmer sådär. Till exempel är p-värdet för test av $\mu = 3430$ mot $\mu \neq 3430$ ca 0.16. Å andra sidan såg vi ju att det fanns lite ”för många” mycket små barn och att vårt normalfördelningsantagande inte var helt klockrent. Med det i åtanke är det ju inte förvånande att medelvärdet blir lite mindre än medianen. Tittar vi stället bara på data för icke tidigt födda barn får vi median 3480 g och medelvärde 3496 g, dvs betydligt närmare varandra.

Notera att teckentestet bara tar hänsyn till hur många observationer som liggar över respektive under m_0 och inte hur långt från m_0 de är. Detta är som det ska vara, eftersom medianen m i sig själv inte tar hänsyn till hur fördelningen ser ut på de två sidorna av m, bara att sannolikhetsmassan är $1/2$ på vardera sidan.

Antag nu att vi vet att data kommer från en symmetrisk fördelning. Då gäller $m = \mathbb{E}[X] = \mu$, så test av median blir också test av väntevärdet och saken kommer i ett annat läge, eftersom observationernas precisa värden är av stor betydelse för vad vi tror om μ; vi brukar ju skatta μ med medelvärdet. Detta kan utnyttjas på följande sätt. Vi vill testa $H_0 : \mu = \mu_0$ mot $H_A : \mu \neq \mu_0$. Rangordna de observerade avvikelserna från μ_0, dvs talen $|X_1 - \mu_0|, \ldots, |X_n - \mu_0|$, i storleksordning från minst till störst. Låt $R_k = j$ om $|X_k - \mu_0|$ är det j:te minsta av dessa tal. Gör detta för alla k, så att varje observation får sin rang. Bilda nu teststatistikan

$$W := \sum_{k:X_k > \mu_0} R_k$$

dvs rangsumman för de observationer som överstiger μ_0. Denna teststatistika kommer att ta hänsyn till om obervationerna på ena sidan μ_0 avviker mer från μ_0 än de på den andra sidan. Värdemängden för W är $0, 1, \ldots, n(n+1)/2$. Om μ_0 är det sanna väntevärdet kan man ganska lätt inse att $\mathbb{E}[W] = n(n+1)/4$ och att frekvensfunktionen är symmetrisk kring detta värde. Mer precist kan man visa att, om $\mu = \mu_0$,

$$\mathbb{P}(W = r) = \frac{a(r)}{2^r},$$

där $a(r)$ är koefficienten framför s^r i utvecklingen av $\prod_{j=1}^{n}(1 + s^j)$. Låt F vara fördelningsfunktionen för en stokastisk variabel med denna fördelning. För att testa H_0 mot H_A, finn c så att $F(n(n+1)/4 + c) \geq 1 - \alpha/2$ och förkasta H_0 på sginifikansnivå högst α om

$$\left| W - \frac{n(n+1)}{4} \right| > c.$$

Detta rangtest går under namnet Wilcoxon Signed Rank Test (WSRT). Koefficienterna $a(r)$ är ganska besvärliga att uttrycka explicit. Om n är stort avhjälps detta till stor del av att det finns en central gränsvärdessats för W. Man kan visa att under H_0 är $\mathbb{V}\mathrm{ar}[W] = n(n+1)(2n+1)/24$ och att

$$\frac{W - n(n+1)/4}{\sqrt{n(n+1)(2n+1)/24}} \approx \mathrm{N}(0,1).$$

Har man tillgång till Matlab har man hursomhelst inga beräkningsproblem. Kommandot heter `signrank`. **Kolla upp** hur det fungerar och upprepa testerna på `fv` och jämför med teckentestet ovan. Du kommer att se att p-värdena för rangtestet blir konsekvent lägre än för teckentestet. Rangtestet är alltså starkare än teckentestet, men kräver i gengäld ett symmetriantagande som teckentestet inte kräver. Jämför också med p-värden för t-testen ovan under normalfördelningsantagande. Har bör du se att rangtestet oftast (men inte alltid) kommer till korta vid jämförelsen.

Låt oss nu titta på ett sätt att göra en ickeparametrisk jämförelse mellan två stickprov. Vi ska anta en *translationsmodell*. Vi har en fördelning F_1 och en fördelning F_2. Vi antar att vi vet att F_1 och F_2 har samma form, men möjligen olika läge, dvs vi vet att det finns ett tal t så att $F_1(x+t) = F_2(x)$ för alla x.

Vi vill testa $H_0 : F_1 = F_2$, (vilket, enligt antagandet om samma form, är samma sak som att testa om $\mu_1 = \mu_2$, eller $t = 0$ om du vill), mot $H_A : F_1 \neq F_2$. Till vårt förfogande har vi två oberoende stickprov, $X_1, \ldots, X_m$ från F_1 och $Y_1, \ldots, Y_n$ från F_2. Antag också att $m \leq n$ (bara en beteckningsfråga). Om nollhypotesen är sann, kan vi se hela samlingen av data, $X_1, X_2, \ldots, X_m, Y_1, Y_2, \ldots, Y_n$ som ett stickprov av storlek $m+n$ på samma fördelning. En konsekvens av det är att om man tittar på hur data ordnar sig storleksmässigt, så bli alla $(m+n)!$ olika permutationer lika sannolika. Låt nu teststatistikan W ges av

$$W := \sum_{k=1}^{m} r(X_k),$$

där $r(X_k)$ är rangen av X_k i det samlade stickprovet. Om H_0 är sann, är den förväntade rangen av en given datapunkt $(m+n+1)/2$, så $\mathbb{E}[W] = m(m+n+1)/2$ och fördelningen för W är symmetrisk kring detta tal. Låt F vara fördelningen för W under H_0. Analogt med tidigare, välj c så att $F(m(m+n+1)/2 + c) \leq \alpha/2$ och förkasta $H_0 : F_1 = F_2$ till förmån för $H_A : F_1 \neq F_2$ på signifikansnivå högst α om

$$\left| W - \frac{m(m+n+1)}{2} \right| > c.$$

Testet kallas för Wilcoxon Rank Sum Test (WRST). Att beräkna F för hand är knöligt, men inget problem för Matlab. Det finns också en central gränsvärdessats även här, analogt med för WSRT ovan, se boken sidan 373. Matlabs kommando heter `ranksum`. **Gör nu** ett sådant rangtest för att se om födelsevikterna tycks skilja sig mellan barn till rökare och barn till ickerökare. Jämför med tvåstickprovs t-testet du gjorde förut. (Notera att när vi gör rangtestet antar vi att de två fördelningarna har samma form, men det gör vi å andra sidan oftast under normalfördelningsantagandet också.)

Diskussion. Generellt kan man säga att parameterfria modeller är bättre än parametermodeller, t.ex. normalfördelningsmodeller, i det att man gör mycket få antaganden om observationernas fördelning. Å andra sidan har oftast de tester man baserar på parametermodeller betydligt bättre styrka, dvs chansen att upptäcka avvikelser från nollhypotesen är betydligt större. Båda har alltså kraftiga fördelar och båda har en stor och viktig plats i tillämpad statistik.

8 Slumpvandringar på grafer

En *graf* är en samling *noder* tillsammans med en samling *kanter* mellan vissa av noderna. En bra bild är att tänka sig en graf som en samling knutpunkter (noderna) med vägar (kanterna) mellan somliga av knutpunkterna. Det är oerhört vanligt med naturligt förekommande fenomen som kan modelleras med grafer, till exempel vägnät, kollektivtrafikkartor, telenät, molekylmodeller, sökträd, sociala relationsnät, phylogenetiska träd, etc. Formellt brukar man beteckna en graf med

$$G = (V, E)$$

där G är namnet på hela grafen, V är mängden av noder (*en*. vertices) och E är mängden av kanter (*en*. edges). Mängden E är en delmängd av mängden av alla par av noder. Detta tolkar man som att om $u, v \in V$ och $\{u, v\} \in E$, så finns det i grafen en kant mellan noderna u och v. Notera att det, enligt denna definition, inte kan förekomma några kanter från en nod till sig själv eller mer än en kant mellan samma par av noder. Det är inte svårt att göra en modell som tillåter även detta. Man brukar då tala om *multigrafer*, men vi har här inget behov av det.

Här ska vi endast titta på ändliga grafer, dvs grafer där V är en ändlig mängd (även om det i andra sammanhang kan vara mycket intressant att studera oändliga grafer).

För $u \in V$, definiera u:s *gradtal*, d_u, som antalet grannar som u har, dvs

$$d_u = |\{v \in V : \{u, v\} \in E\}|.$$

Här har vi använt den vanligt förekommande beteckningen $|A|$ för antalet element i en mängd A. En *slumpvandring* på en graf G är den Markovkedja som beskrivs av en partikel som hoppar mellan noder på ett sådant sätt att den vid varje hopp på måfå väljer en av grannarna till den nod där den nu står. Mer formellt är slumpvandringen den Markovkedja $X_0, X_1, X_2, \ldots$ med V som tillståndsrum, som rör sig så att det för alla $t = 0, 1, 2, \ldots$ och alla $u \in V$ gäller att

$$\mathbb{P}(X_{t+1} = v | X_t = u) = \frac{1}{d_u}$$

för alla v sådana att $\{u, v\} \in E$.

Låt oss nu generalisera det vi gjort en aning. Istället för att betrakta en graf, ska vi betrakta en *viktad* graf, dvs en graf $G = (V, E)$, där det till varje kant $\{u, v\} \in E$ finns en *vikt*, dvs ett ickenegativt tal w_{uv}. Detta kan till exempel användas i modellen för ett vägnät, där man vill använda vikterna till att markera hur stora vägarna är. Motsvarigheten till gradtalet d_u för en nod i en (oviktad) graf, blir här *nodens vikt*, w_u, som definieras som

$$w_u = \sum_{v:\{u,v\}\in E} w_{uv}.$$

En slumpvandring skiljer sig från en slumpvandring på den oviktade grafen endast genom att vi nu istället låter

$$\mathbb{P}(X_{t+1} = v | X_t = u) = \frac{w_{uv}}{w_u}.$$

Man ser att en slumpvandring på en oviktad graf fås som specialfallet där man låter $w_{uv} = 1$ för alla $\{u, v\} \in E$ (notera att man får $w_u = d_u$ för alla u).

När man talar om slumpvandring på en viktad graf, är det lätt att inse att man faktiskt alltid kan betrakta grafen som *fullständig*, dvs att mellan *varje* par av noder finns en kant. Detta kan man göra eftersom, om man då får en kant som egentligen inte skulle funnits, kan man helt enkelt kompensera för det genom att ge den kanten vikten 0.

Uppgift: Skriv ett Matlabprogram som simulerar slumpvandring på några olika viktade grafer. Försök, genom att köra simuleringarna tillräckligt länge, att bilda dig en uppfattning om vad den stationära fördelningen är. Du väljer själv antal noder, startfördelning, vikter och hur många steg du simulerar och hur du varierar dessa parametrar för att lösa uppgiften. Skriv gärna ett program som tar dessa parametrar som indata.

När du har din uppfattning klar, försök att bevisa den. Tips: slumpvandringen utgör en reversibel Markovkedja.

Anmärkning. Faktum är att det, rent programmeringsmässigt, är lättare att explicit beräkna fördelningen för X_t snarare än att simulera. Kom ihåg att om P är Markovkedjans övergångsmatris och

$$\mathbf{p}_t = [\mathbb{P}(X_t) = 1\, \mathbb{P}(X_t = 2) \ \ldots \ \mathbb{P}(X_t = n)]$$

gäller att

$$\mathbf{p}_t = \mathbf{p}_0 P^t.$$

Låt nu bara Matlab beräkna P utifrån vikterna och mata in i denna beräkning för ett stort t.