Labbrapport i Statistik

Laboration 4

732g43

Jakob Lindén

Avdelningen för Statistik och maskininlärning
Institutionen för datavetenskap
Linköpings universitet
06-01-2020

Innehåll

1. Uppgift 1

I följande uppgift kommer ett datamaterial om vaccinerade barn i ett latinamerikanskt land att användas. Syftet är att förklara sannolikheten för att ett barn är vaccinerat mot en sjukdom. Icke-informativa priorfördelningar kommer användas. Responsvariabeln $y_i = 1$ om ett barn i är vaccinerat mot sjukdomen och 0 annars. Följande förklaringsvariablerna finns att tillgå för att förklara sannolikheten att ett barn är vaccienerat:

$x_1 = 1$ om barnet är minst 2 år, 0 annars.

$x_2 = 1$ om barnet bor på landsbygden, 0 annars.

$x_3 = 1$ om modern till barnet har genomgått endast första steget av elementär grundskoleutbildning 0 annars.

$x_4 = 1$ om modern till barnet har genomgått både det första och andra steget av elementär grundskoleutbildning, 0 annars (andra steget förutsätter att det första steget är genomgått).

$x_5 =$ andel procent infödda i det samhälle som barnet bor i.

De icke-informativa priorfördelningarna för parametrarna som kommer användas i denna uppgift är följande:

$$\begin{aligned} \beta_0 &\sim N(0, 10) \\ \beta_j &\sim N(0, 10) \end{aligned} \tag{1.1}$$

1.1 a)

```
require(rethinking)
require(xtable)


load("C:/Users/Jakob/Desktop/Bayesian statistics/lab4/VaccinationsData.Rdata")

#### Uppgift 1 ####
#### a)

Upg1_data <- Vaccination_Barn[,1:3]

flist <- alist( y ~ dbinom( 1 , p ) ,
                logit(p) <- b0 + b1*x1 + b2*x2,
                b0 ~ dnorm(0,10) ,
                b1 ~ dnorm(0,10) ,
                b2 ~ dnorm(0,10)
)

Modell_1a <- map2stan(flist,data=Upg1_data)

#Konvergens analys
```

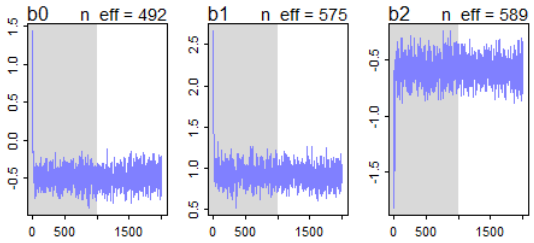
```
asd<-precis(Modell_1a, prob = 0.909)

xtable(asd@output,label = "upg1a")
asd@output

tracerplot(Modell_1a)

PostSamp_1a <- extract.samples(Modell_1a)

par(mfrow=c(2,2))
#b0
AnvPar <- PostSamp_1a$b0
NIter <- length(PostSamp_1a$b0)
Means <- matrix(0,nrow=NIter,ncol=1,byrow=TRUE)
for (iter in 1:NIter){
  Means[iter] <- mean(AnvPar[1:iter])
}
plot(Means, main = "Konvergens för b0 posteriormean")

#b1
AnvPar <- PostSamp_1a$b1
NIter <- length(PostSamp_1a$b1)
Means <- matrix(0,nrow=NIter,ncol=1,byrow=TRUE)
for (iter in 1:NIter){
  Means[iter] <- mean(AnvPar[1:iter])
}
plot(Means, main = "Konvergens för b1 posteriormean")

#b2
AnvPar <- PostSamp_1a$b2
NIter <- length(PostSamp_1a$b2)
Means <- matrix(0,nrow=NIter,ncol=1,byrow=TRUE)
for (iter in 1:NIter){
  Means[iter] <- mean(AnvPar[1:iter])
}
plot(Means, main = "Konvergens för b2 posteriormean")
```

En Bayesiansk logistisk regression med y som responsvariabel och förklaringsvariablerna x_1 och x_2 ska anpassas med MCMC-algoritmen. Slutsatser ska dras om MCMC dragningarna lett till att algoritmen konvergerat till posteriorn med hjälp av n_{eff}, $\hat{R}$ samt tracerplottar och plottar för varje parameters ackumulerade posteriormedelvärde över MCMC dragningarna. Till att börja med undersöks n_{eff} och $\hat{R}$ i tabell 1.1. Det ses att enligt dessa två mått har modellen konvergerat eftersom n_{eff} är större än 100 för samtliga parametrar och $\hat{R}$ är godtyckligt nära 1 för alla parametrar.

Tabell 1.1

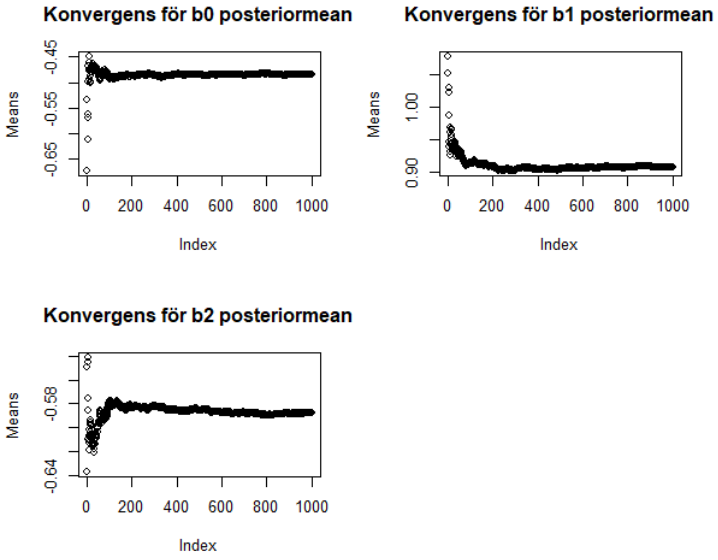
	Mean	StdDev	lower 0.909	upper 0.909	n_eff	Rhat
b0	-0.48	0.12	-0.71	-0.30	491.65	1.01
b1	0.91	0.11	0.73	1.10	575.28	1.00
b2	-0.59	0.10	-0.76	-0.41	588.60	1.00

I figur 1.1 ses en tracerplott där det observeras att det är städade markov kedjor som producerats, både stationära och väl mixade med zikzak mönster.



Figur 1.1: Tracerplott

För att undersöka om parametrarna har konvergerat till något specifikt värde vilket är önskat underöks figur 1.2. Det ses att β_0 har konvergerat till ungefär -0.50 och β_1 konvergerat till 0.90 över posteriordragningarna. Till sist observeras att β_2 konvergerat till ungefär -0.60.



Figur 1.2: Parameter konvergens

Med dessa mått i beaktning kan vi dra slutsatsen att MCMC dragningarna lett till att MCMC algoritmen konvergerat till posteriorn.

1.2 b)

```
#### b)

binom_df <- data.frame("y" = rep(0,times=4),"x1" = c(0,1,0,1), "x2" = c(0,0,1,1),"Unsuccesfull"
    = rep(0,times=4))
binom_df[1,1] <- sum(Upg1_data[Upg1_data$x1 == 0 & Upg1_data$x2 == 0,1])
binom_df[2,1] <- sum(Upg1_data[Upg1_data$x1 == 1 & Upg1_data$x2 == 0,1])
binom_df[3,1] <- sum(Upg1_data[Upg1_data$x1 == 0 & Upg1_data$x2 == 1,1])
binom_df[4,1] <- sum(Upg1_data[Upg1_data$x1 == 1 & Upg1_data$x2 == 1,1])
binom_df[,4] <- aggregate(x = Upg1_data$y,
                by = list(x1 = Upg1_data$x1, x2 = Upg1_data$x2),
                FUN = length)[,3]

flist <- alist( y ~ dbinom(Unsuccesfull,p) ,
```

```
                    logit(p) <- b0 + b1*x1 + b2*x2,
                    b0 ~ dnorm(0,10) ,
                    b1 ~ dnorm(0,10) ,
                    b2 ~ dnorm(0,10)
)
Modell_binom <- map2stan(flist,data=binom_df)
```

Liknande modell som i föregående uppgift (a) ska anpassas fast denna gång med Bayesiansk binomial regression. Detta för att verifiera att den logistiska och binomiala regressionen ger samma resultat. Modellen anpassas och resultatet redovisas i tabell 1.2. Det observeras att modellerna produceras exakt samma resultat vad det gäller medelvärde och standardavvikelse för parametrarna. Små skillnader observeras på tredje decimalen i kredibilitetsintervallet men till allra största grad producerar modellerna som väntat samma resultat.

Tabell 1.2

	Mean	StdDev	lower 0.909	upper 0.909	n_eff	Rhat
b0	-0.48	0.12	-0.69	-0.28	459.68	1.00
b1	0.91	0.11	0.73	1.11	519.24	1.00
b2	-0.59	0.10	-0.76	-0.42	528.08	1.00

1.3 c)

```
### c)

p.x10.x20 <- as.vector(logistic( PostSamp_1a$b0 ))
binom_df[1,1]/binom_df[1,4]
p.x11.x20 <- as.vector(logistic( PostSamp_1a$b0 + PostSamp_1a$b1 ))
binom_df[2,1]/binom_df[2,4]
p.x10.x21 <- as.vector(logistic( PostSamp_1a$b0 + PostSamp_1a$b2 ))
binom_df[3,1]/binom_df[3,4]
p.x11.x21 <- as.vector(logistic( PostSamp_1a$b0 + PostSamp_1a$b1 + PostSamp_1a$b2 ))
binom_df[4,1]/binom_df[4,4]

precis(p.x10.x20, prob = 0.952)
precis(p.x11.x20, prob = 0.952)
precis(p.x10.x21, prob = 0.952)
precis(p.x11.x21, prob = 0.952)

plot(x = 1:4,y = c(0.38,0.6,0.26,0.46), type = "p", ylim = c(0.2,0.7),ylab="",xlab="")
lines(x = c(1,1),y = c(0.32,0.44))
lines(x = c(2,2),y = c(0.56,0.64))
lines(x = c(3,3),y = c(0.22,0.29))
lines(x = c(4,4),y = c(0.43,0.48))
points(x = 1:4,y = c(0.39,0.6,0.25,0.46), col = rgb(red = 0, green = 0, blue = 1, alpha = 0.5),
    pch=19)
```

Modellen i uppgift (a) ska utvärderas med avseende på hur bra den anpassar sannolikheten p för att ett barn är vaccinerat mot sjukdomen enligt dessa mått: ett 95.2 procent kredibilitetsintervall för p för varje kombination av värden på x_1 och x_2, dessa andelar ska jämföras mot de sanna andelarna i data.

Sannolikheten att ett barn är vaccinerat givet att barnet är yngre än 2 år och att barnet inte bor på landsbygden ses i modellen med 95.2 procent sannolikhet ligga mellan 32 och 44 procent, detta ses i tabell 1.3. Den 'sanna' sannolikheten från data är 39 procent, således verkar modellen anpassa data bra i detta fall.

Tabell 1.3: $x_1 = 0$ och $x_2 = 0$

	Mean	StdDev	|0.952	0.952|
model	0.38	0.03	0.32	0.44

Sannolikheten att ett barn är vaccinerat givet att barnet är minst 2 år och att barnet inte bor på landsbyggden ska undersökas. Krediblitetsintervallet påvisar att denna sannolikhet ligger mellan 56 och 65 procent med 95.2 procent sannolikhet. Från data ses det att sannolikheten är 60 procent, således dras slutsatsen att modellen verkar anpassa data bra.

Tabell 1.4: $x_1 = 1$ och $x_2 = 0$

	Mean	StdDev	|0.952	0.952|
model	0.60	0.02	0.56	0.65

Sannolikheten att ett barn är vaccinerat givet att barnet är yngre än 2 år och att barnet bor på landsbyggden ska undersökas. Kredibilitetsintervallet påvisar att med 95.2 procent sannolikhet ligger denna sannolikhet mellan 22 och 29 procent. Vid undersökning av den sanna sannolikheten ses denna vara 25 procent, således dras slutsatsen att modellen verkar anpassa data bra.

Tabell 1.5: $x_1 = 0$ och $x_2 = 1$

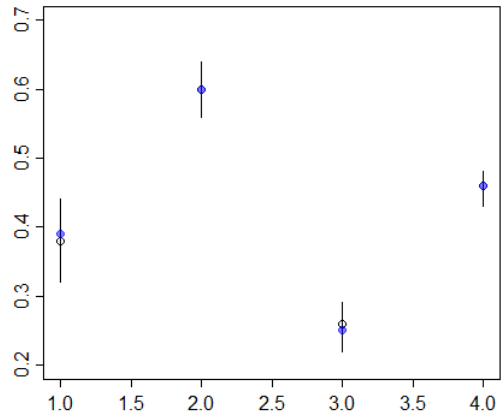
	Mean	StdDev	|0.952	0.952|
model	0.26	0.02	0.22	0.29

Slutligen undersöks sannolikheten att ett barn är vaccinerat givet att barnet är minst 2 år samt att barnet bor på landsbyggden. Enligt kredibilitetsintervallet ses denna sannolikhet ligga mellan 43 och 49 procent med 95.2 procent sannolikhet. Enligt data är den 'sanna' sannolikheten 46 procent, således dras slutsatsen att modellen anpassar data bra.

Tabell 1.6: $x_1 = 1$ och $x_2 = 1$

	Mean	StdDev	|0.952	0.952|
model	0.46	0.01	0.43	0.49

I figur 1.3 ses de 95.2 procent kredibilitetsintervallen för de fyra kombinationerna samt posteriormedelvärdet som vit prick och det sanna värdet från data som blå prick. Det ses i figuren att modellen anpassar data väldigt bra eftersom posteriormedelvärdet för respektive kombination av förklaringsvariablerna genererar samma sannolikhet som anses vara den 'sanna' sannolikheten i data.



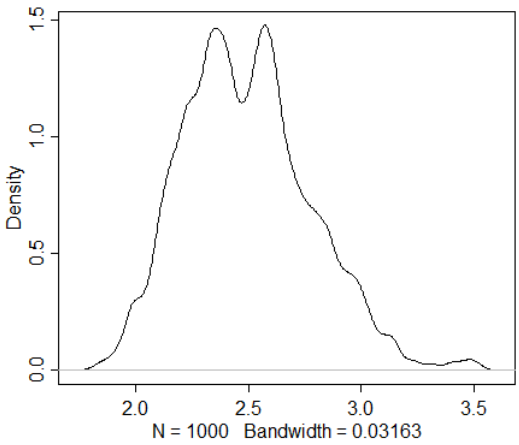
Figur 1.3: Posterior mean jämfört mot data

1.4 d)

```
### d)

ExpChangeOdds <- exp(PostSamp_1a$b1)
dens(ExpChangeOdds)
```

Posteriorfördelningen för den faktor som oddset för ett vaccinerat barn förändras med då man går från ett barn under 2 år till ett barn på minst 2 år ska redovisas i en plott.

Posteriorfördelningen för oddset redovisas i figur 1.4 där det ses att fördelningen har mest massa ungefär vid 2.5. Således dras slutsatsen att för ett barn som är minst två år är det ungefär 2.5 gånger så mer sannolikt att detta barnet är vaccinerat gentemot om ett barn som är under 2 år.



Figur 1.4

1.5 e)

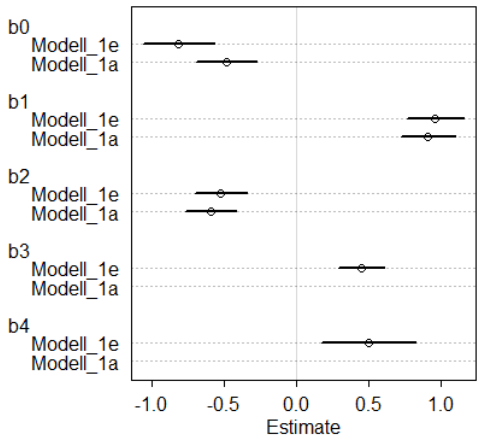
```
### e)
```

```
Upg1e_data <- Vaccination_Barn[,1:5]

flist <- alist( y ~ dbinom( 1 , p ) ,
                logit(p) <- b0 + b1*x1 + b2*x2 + b3*x3 + b4*x4,
                b0 ~ dnorm(0,10) ,
                b1 ~ dnorm(0,10) ,
                b2 ~ dnorm(0,10) ,
                b3 ~ dnorm(0,10) ,
                b4 ~ dnorm(0,10)
)

Modell_1e <- map2stan(flist,data=Upg1e_data)

DIC(Modell_1a)[1]
DIC(Modell_1e)[1]


plot( coeftab(Modell_1a,Modell_1e),prob=0.909 )
```

Informationskriteriet DIC ska användas för att avgöra om det är bättre att lägga till förklaringsvariablerna x_3 och x_4 om moderns utbildningsnivå till modellen i uppgift (a).

Modellen anpassas och DIC värdena beräknas till 2871 för modellen i uppgift (a) och då variabler om moderns utbildningsnivå läggs till beräknas DIC till 2850. Således dras slutsatsen att modellens prediktiva förmåga blir bättre med x_3 och x_4 i tillagda i modellen med avseende på DIC som utvärderings mått.

I figur 1.5 ses förändringen i parametrarnas kredibilitetsintervall (90.9 procentigt) då x_3 och x_4 läggs till i modellen. Det verkar inte kunna påvisas nån confounding effekt då dessa variabler läggs till, detta eftersom kredibilitetsintervallen för repsketive parameter förändras nämnvärt då x_3 och x_4 läggs till i modellen. Den största förändringen mellan modellerna är för β_0, dock ses denna parameter vara entydigt linjärt negativ i båda modellerna, således kan det inte påstås skett nån confounding effekt.



Figur 1.5

1.6 f)

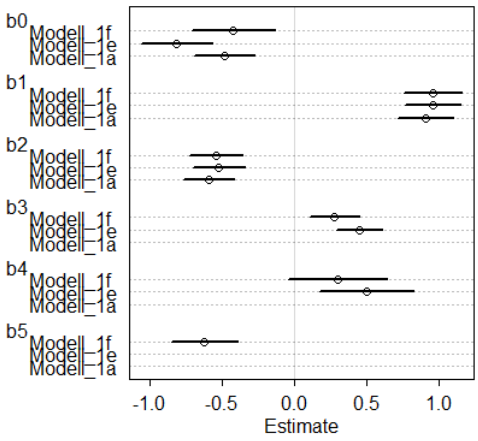
```
### f)

Upg1f_data <- Vaccination_Barn[,1:6]
```

```
flist <- alist( y ~ dbinom( 1 , p ) ,
               logit(p) <- b0 + b1*x1 + b2*x2 + b3*x3 + b4*x4 + b5*x5,
               b0 ~ dnorm(0,10) ,
               b1 ~ dnorm(0,10) ,
               b2 ~ dnorm(0,10) ,
               b3 ~ dnorm(0,10) ,
               b4 ~ dnorm(0,10) ,
               b5 ~ dnorm(0,10)
)

Modell_1f <- map2stan(flist,data=Upg1f_data)

#DIC
DIC(Modell_1a)[1]
DIC(Modell_1e)[1]
DIC(Modell_1f)[1]
#WAIC
WAIC(Modell_1a)[1]
WAIC(Modell_1e)[1]
WAIC(Modell_1f)[1]
#Confounding check
plot( coeftab(Modell_1a,Modell_1e,Modell_1f),prob=0.909 )
#Kredbilitetsband
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
modex1 <- getmode(Upg1f_data$x1)
modex2 <- getmode(Upg1f_data$x2)
modex3 <- getmode(Upg1f_data$x3)
modex4 <- getmode(Upg1f_data$x4)

plot( Upg1f_data$y ~ Upg1f_data$x5 ,ylim=c(0,1.5),type="n" )

PostSamp_1f <- extract.samples(Modell_1f)
x5.seq <- seq(from=min(Upg1f_data$x5),to=max(Upg1f_data$x5),by=0.01)
mu.ci <- sapply( x5.seq , function(x)
  PI( exp( PostSamp_1f$b0 + PostSamp_1f$b1*modex1 + PostSamp_1f$b2*modex2 +
            PostSamp_1f$b3*modex3 + PostSamp_1f$b4*modex4 + PostSamp_1f$b5*x) , prob=0.952) )
shade( mu.ci , x5.seq )
```

Nu ska en Bayesiansk logistisk regressions modell med y som responsvariabel och med alla förklaringsvariabler anpassas. Utifrån DIC och WAIC måtten ska slutsats dras om vilken av modellern i uppgift (a),(e) eller (f) som är bäst. Efter det ska confounding effekter undersökas då x_5 läggs till i modellen. Slutligen ska även ett 95.2 procent kredibilitetsintervall (kredibilitetsband) beräknas för sannolikheten p att ett barn är vaccinerat som funktion av förklaringsvariabeln x_5 genom att använda typvärdet för respektive x_1,x_2,x_3 och x_4 i datamaterialet.

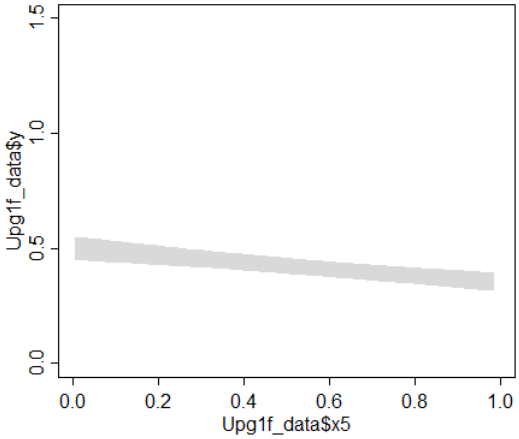
Till att börja med undersöks DIC värdena för respektive modell. DIC beräknas till 2871, 2850 respektive 2829 för modell a, e respektive f. Modellen med alla förklaringsvariabler påvisar således bäst prediktiv förmåga med avseende på DIC, detta trots att DIC är straffande för antalet parametrar. WAIC för de tre modellerna beräknas till 2871, 2850 och 2829, således avrundat till exakt samma som DIC. Samma slutsats dras således att den fulla modellen har bäst prediktiv förmåga med avseende på både DIC och WAIC som informationskriterium.

För att undersöka om det har skett några confounding effekter då x_5 lagts till i modellen undersöks figur 1.6. Det har skett en tydlig confounding effekt och det är på parametern β_4 som då variabeln x_5 läggs till i modellen inte längre är entydigt linjärt positivt vilket anses vara en confounding effekt.



Figur 1.6

Slutligen skapas ett 95.2 procent kredilitetsintervall för sannolikheten att ett barn är vaccinerat som funktion av förklaringsvariabeln x_5 genom att använda typvärdena för de övriga binära förklaringsvarialberna. Detta ses i figur 1.7 där sannolikheten att ett barn är vaccinerat verkar minska då andelen procent infödda i det samhälle som barnet bor i ökar givet typvärdet på de övriga förklaringsvariablerna. Kredibilitetsbandet påvisar också att osäkerheten är lite större för låga värden på x_5 än höga värden, men väldigt marginellt.



Figur 1.7

2. Uppgift 2

I datamaterialet som användes i uppgift 1 finns det information om index för vilket samhälle som barnet tillhör. Detta kommer användas för att anpassa modeller med flera olika intercept för varje samhälle.

2.1 a)

```
Upg2_data <- Vaccination_Barn[,c(1:5,7)]

flist <- alist( y ~ dbinom( 1 , p ) ,
               logit(p) <- b0[z1] + b1*x1 + b2*x2 + b3*x3 + b4*x4,
               b0[z1] ~ dnorm(0,10) ,
               b1 ~ dnorm(0,10) ,
               b2 ~ dnorm(0,10) ,
               b3 ~ dnorm(0,10) ,
               b4 ~ dnorm(0,10)
)

Modell_2 <- map2stan(flist,data=Upg2_data,iter = 20000)

DIC(Modell_2)[1]
DIC(Modell_1e)[1]
WAIC(Modell_2)[1]
WAIC(Modell_1e)[1]
```

Nu ska skillnader i sannolikhet för vaccination utifrån vilket samhälle som barnet tillhör undersökas. En Bayesiansk logistisk regressions modell med olika intercept för olika samhällen, där y är responsvariabel med x_1,x_2,x_3 och x_4 som förklaringsvariabler.

Denna modell tar inte hänsyn till pooling av information från data mellan samhällena för att anpassa intercepten. Detta eftersom detta inte är en multilevel modell utan här har varje individuellt intercept en egen priorfördelning medans i en multilevel modell där intercepten har en gemensam priorfördelning. Således ges icke-precis information om interceptet i varje grupp eftersom det är endast data i respektive grupp som uppskattar sitt intercept. Detta kan medföra problem då vissa grupper har få datapunkter och då kommer det leda till att modellen överanpassar data med lite information i de grupperna.

MCMC dragningarna ska utvärderas för att se om de lett till att MCMC algoritmen konvergerat med hjälp av n_{eff} och $\hat{R}$. I tabell 2.1 presenteras endast de första fem och sista fem intercepten eftersom det var 161 intercept blev det för mycket att redovisa i tabell. Alla intercept ses ha n_{eff} större än 100 och $\hat{R}$ runt 1, detta eftersom 20 000 MCMC dragningar utförts.

Tabell 2.1

	Mean	StdDev	lower 0.89	upper 0.89	n_eff	Rhat
b0[1]	7.08	6.43	-2.29	16.49	5720.61	1.00
b0[2]	-0.49	0.60	-1.47	0.46	8393.89	1.00
b0[3]	0.48	0.91	-0.97	1.88	7306.70	1.00
b0[4]	0.36	0.90	-1.07	1.73	8023.25	1.00
b0[5]	-0.71	1.81	-3.54	2.18	8609.34	1.00
⋮	⋮	⋮	⋮	⋮	⋮	⋮
b0[157]	0.93	1.15	-0.79	2.88	205.26	1.00
b0[158]	0.48	1.21	-1.40	2.46	239.37	1.00
b0[159]	0.17	1.15	-1.73	1.98	206.13	1.00
b0[160]	1.24	1.26	-0.65	3.42	264.56	1.00
b0[161]	1.88	1.25	-0.10	3.88	249.54	1.00
b1	1.13	0.13	0.92	1.33	4397.69	1.00
b2	-2.18	1.02	-3.90	-0.61	163.10	1.00
b3	0.19	0.12	-0.01	0.39	5668.98	1.00
b4	0.23	0.26	-0.22	0.62	5639.14	1.00

För att undersöka hur denna modell som tar hänsyn till olika intercept för olika samhällen utför prediktioner så undersöks DIC och WAIC mellan denna modell och modellen som anpassades i uppgift (1e). DIC för denna modell beräknades till 2779 och DIC för uppgift (1e) modellen beräknades till 2850. Således dras slutsatsen att med avseende DIC så har modellen som tar hänsyn till olika intercept för olika samhälle bäst prediktiv förmåga. WAIC för denna modell beräknas till 2835 och för modellen i uppgift (1e) beräknas WAIC till 2850. Således dras slutsatsen att med avseende WAIC har modellen som tar hänsyn till olika intercept för olika samhällen bäst prediktiv förmåga.

2.2 b)

```
flist <- alist( y ~ dbinom(1,p) ,
                logit(p) <- a[z1] + b1*x1 + b2*x2 + b3*x3 + b4*x4,
                a[z1] ~ dnorm(a0,sigma_a) ,
                a0 ~ dnorm(0,10) ,
                sigma_a ~ dcauchy(0,10) ,
                b1 ~ dnorm(0,10) ,
                b2 ~ dnorm(0,10) ,
                b3 ~ dnorm(0,10) ,
                b4 ~ dnorm(0,10)
)
Modell_2b <- map2stan(flist,data=Upg2_data, iter = 20000)

DIC(Modell_1e)[1]
DIC(Modell_2)[1]
DIC(Modell_2b)[1]
WAIC(Modell_1e)[1]
WAIC(Modell_2)[1]
WAIC(Modell_2b)[1]
```

Återigen ska skillnader i sannolikhet för vaccination utifrån vilket samhälle som barnen tillhör tas hänsyn till. Denna gång med en multilevel modell. Denna modell ska jämföras med avseende DIC och WAIC mot modellerna i uppgift 1(e) och 2(a). 20000 MCMC dragningar görs och uppnår konvergens med avseende $n_{eff} > 100$ och $\hat{R} \approx 1$, detta ses i tabell 2.2.

Tabell 2.2

	Mean	StdDev	lower 0.909	upper 0.909	n_eff	Rhat
a[1]	-0.63	0.70	-1.75	0.59	3296.69	1.00
a[2]	-0.63	0.46	-1.43	0.14	2814.75	1.00
a[3]	-0.30	0.55	-1.23	0.64	3755.64	1.00
a[4]	-0.33	0.54	-1.29	0.58	2654.51	1.00
a[5]	-0.81	0.66	-1.89	0.32	3591.96	1.00
⋮	⋮	⋮	⋮	⋮	⋮	⋮
a[157]	-0.67	0.47	-1.47	0.14	1486.27	1.00
a[158]	-0.92	0.53	-1.84	-0.05	1822.59	1.00
a[159]	-1.19	0.47	-1.95	-0.39	1404.19	1.00
a[160]	-0.55	0.53	-1.47	0.34	2200.74	1.00
a[161]	-0.23	0.54	-1.08	0.70	1638.42	1.00
a0	-0.80	0.19	-1.11	-0.48	334.46	1.00
sigma_a	0.71	0.09	0.58	0.86	1019.25	1.00
b1	1.01	0.12	0.79	1.20	932.01	1.00
b2	-0.61	0.17	-0.92	-0.33	519.20	1.00
b3	0.34	0.11	0.15	0.52	1804.94	1.00
b4	0.39	0.21	0.03	0.74	1696.28	1.00

Skillnaderna i denna modell jämfört med modellen i uppgift 2(a) är att intercepten i denna multilevel modell har en gemensam priorfördelning. Således uppdateras den gemensamma priorn $\alpha_j \sim N(\alpha, \sigma_\alpha)$ efter data eftersom parametrarna α och σ_α i α_j har egna prior fördelningar. På detta sätt kan information från de grupper med många observationer användas för att poola information till de grupper som har väldigt få observationer. På detta sätt blir anpassningen för modellen som helhet bättre eftersom informationen poolas mellan grupperna. DIC för modellen i uppgift (1e) beräknades till 2850, för modellen i uppgift (2a) till 2778 och slutligen för denna modell till 2740. Således med avseende på DIC har denna multilevel modell bäst prediktiv förmåga.

WAIC för modellen i uppgift (1e) beräknades till 2850, för modellen i uppgift (2a) till 2833 och slutligen för denna modell till 2739. Således med avseende på WAIC har denna multilevel modell bäst prediktiv förmåga. Som slutsats ses mutlilevel modellen alltså vara den bästa modellen och ha bäst prediktiv förmåga. Detta var att förvänta eftersom vissa samhällen har väldigt få observationer och då är det väldigt att kunna använda information från andra samhällen för att anpassa dessa bra.

3. Uppgift 3

Data om antalet flygbombsträffar i London ska användas i följande uppgift.

3.1 a)

```
bombman <- data.frame(pois = c(rep(0, 229), rep(1, 211), rep(2, 93), rep(3, 35), rep(4, 7),
    rep(5, 1)))

#Gamma(alpha,beta) konjugerade prior för poisfördelning
#Icke-inf prior -> Gamma(1 + sumxi, 1 + n)
flist <- alist( pois ~ dpois(lambda) ,
                        lambda <- dgamma(1,1)
)
resBomb <- map(flist, data=bombman)

bombSamples <- extract.samples(resBomb, n = 1e4)

lambda_posterior <- bombSamples$lambda

mean(lambda_posterior)
sd(lambda_posterior)
```

En icke-informativ konjugerad prior ska användas för medelvärdet θ och parametervärdena för priorn ska redovisas. 10000 värden från posteriorfördelninge ska simuleras med hjälp av kvadratisk approximation. Posteriorfördelningens medelvärde och standardavvikelse ska beräknas och jämföras med värdena från föreläsningen på moment 1.

Gammafördelningen är en konjugerad prior till poissonfördelningen, där posteriorfördelningens hyperparametrar blir $\alpha + \sum_{i=1}^{n} x_i, \beta + n$. När vi undersöker posteriorfördelningens hyperparametrar ser vi att en icke-informativ prior innebär låga värden på α och β eftersom då baseras posteriorfördelningens fördelning till störst del av data. Således väljs en gammafördelning med $\alpha = 1$ och $\beta = 1$ som priorfördelning till θ. En kvadratisk approximation anpassas och 10000 värden dras från posteriorfördelningen. Medelvärdet och standardavvikelsen för $\bar{y}$ och σ är 0.927 respektive 0.04. I föreläsningen från moment 1 löstes detta analytiskt där resultatet blev 0.932 för $\bar{y}$ respektive 0.04 för standardavvikelsen σ, alltså väldigt lika resultat. Resultaten stämmer överens eftersom den kvadratiska approximationen gör en bra anpassning där den icke-informativa priorn i båda fallen låter data väga tungt, således blir resultatet väldigt lika med en liten avvikelse för det görs fortfarande slumpmässigt och resultaten kommer aldrig bli exakt analytiskt som med simulering.

3.2 b)

```
flist <- alist( pois ~ dpois(lambda) ,
                        log(lambda) <- b0,
                         b0 ~ dnorm(0,10)
)
```

```
resBomb_2b <- map2stan(flist, bombman)

bombSamples_2b <- extract.samples(resBomb_2b,n=1e3)

sum(exp(bombSamples_2b$b0) > 1)/length(bombSamples_2b$b0)
```

Modellen för Poisson regression från föreläsningen på moment 4 ska användas men utan förklaringsvariabler. Modellen ska anpassas med MCMC och posteriorsannolikheten för medelvärdet $\lambda : P(\lambda > 1|y)$ ska beräknas.

Modellen anpassas och sannolikheten för att λ ska anta värden över 1 beräknas till 3.6 procent.