INLA Extended Examples

INLA

- Integrated Nested Laplace Approximation
  - Posterior approximation which relies on numerical integration and sparse matrix analysis
  - Is particularly suited to Gaussian models, especially if the effects are log-Gaussian or Gaussian
  - Linear Mixed Models Or
  - Generalized linear mixed models

Model fitting: McMC and INLA

- Conventionally Markov chain Monte Carlo is used to estimate posterior quantities for Bayesian models (such as the convolution or log-normal models)
  - WinBUGS is designed to do this via two basic methods
    - Gibbs sampling
    - Metropolis –Hastings
- Approximation to posterior distributions has recently become available via Laplace approximation in the INLA package
  - Does not require iterative computation (unlike McMC)
  - Fast computation

Models on INLA

- INLA operates as for the LM function on R
  - Two components:
    - formula and inla call

- Example:

>formula1=y~1+x

>result1=inla(formula1, family="gaussian",data='dataframe')

- This fits a linear regression with intercept between y and x

Linear Regression models

- A simple model:

$$y_i = \alpha_0 + \alpha_1 x_{1i} + e_i$$

$$e_i \sim N(0, \tau_y^{-1}); \qquad \tau_y^{-1} = \sigma_y^2$$

$$\alpha_* \sim N(0, \tau_{\alpha_*}^{-1})$$

with two predictors:

$$y_i = \alpha_0 + \alpha_1 x_{1i} + \alpha_2 x_{2i} + e_i$$

INLA Basics: Regression (file: INLA_basic_regression.txt)

- Bayesian linear regression example : y and 2 predictors (x1,x2)

```
x1<-c(1.1,2.3,3.4,4.5,5.4)
x2<-c(-2.3,4.5,3.6,6.8,12.7)
y<-c(1.2,1.4,2.3,3.2,1.2)
As<-data.frame(x1,x2,y)
```

Model Specification

```
library(INLA)
```

regression model formula (one predictor)

```
formula1<-y~1+x1

#model fitting
res1<-inla(formula1,family="gaussian",
data=As,control.compute=list(dic=TRUE))
```

Output

```
summary(res1)
```
#summary of output

```
sum1<-res1$summary.fixed
```
stored the regression estimates

```
res1$dic
```
display the DIC

New 2 predictor Model

```
formula2<-y~1+x1+x2

res2<-inla(formula2,family="gaussian",
data=As,control.compute=list(dic=TRUE))
```

Use of f() function

- A powerful feature of the INLA package is the f() function

- This allows special links to be specified to predictors
  - Can have smooth non-linear links
  - Can have correlated dependence
  - Can include random effects via this function

Examples

1) *random intercept model* (uncorrelated random effect)

```
ind<-seq(1:5)
formula3<-y~1+x1+f(ind,model='iid')
```

$$\boxed{\begin{aligned} y_i &= \alpha_0 + \alpha_1 x_{1i} + v_i + e_i \\ v_i &\sim N(0, \tau_v^{-1}) \end{aligned}}$$

2) *factor effect with random intercept model*

```
ind2<-c(1,1,1,2,2)
formula4<-y~1+x1+f(ind,model='iid')+as.factor(ind2)
```

Random slope model and smoothed random walk model

$$y_i = \alpha_0 + \alpha_1 x_{1i} + \alpha_{2i} x_{2i} + e_i$$

$$\alpha_{2i} \sim N(0, \tau_{\alpha_2}^{-1})$$

Formula5<-y~1+x1+f(ind,x2,model="iid")

$$y_i = \alpha_0 + \alpha_1 x_{1i} + f(x_{2i}) + e_i$$

$f()$: random walk smoothing prior

Formula6<-y~1+x1+f(x2,model="rw1")

Worked Example of Model 5 Random slope and fixed effect

```
formula5<-y~1+x1+f(ind,x2,model="iid")
res5<-inla(formula5,family="gaussian",data=As,
control.compute=list(dic=TRUE,cpo=TRUE))
```

Results 1

summary(res5)

Fixed	effects:				
	mean	sd	0.025quan	0.5quant	0.975quan
(Intercept)	0.7152	0.3126	0.077	0.7152	1.3536
x1	0.426	0.1295	0.1617	0.426	0.6904

Results 2

Deviance Information Criterion: -29.11			
Effective number of parameters: 5.61			
Marginal Likelihood: -13.83			

Results 3

fixed<-res5$summary.fixed

	mean	sd	0.025quan	0.5quant	0.975quan	kld
(Intercept)	0.715224	0.312635	0.076967	0.715218	1.35364	0.00E+00
x1	0.426023	0.129473	0.161689	0.426017	0.690434	4.04E-19

Results 4

random<-res5$summary.random

$ind							
	ID	mean	sd	0.025quan	0.5quant	0.975quan	kld
1	1	-0.00701	0.088193	-0.18709	-0.00701	0.173133	2.13E-22
2	2	-0.06551	0.035798	-0.13861	-0.0655	0.007545	0.00E+00
3	3	0.037793	0.064662	-0.09421	0.037788	0.169884	8.86E-20
4	4	0.083446	0.0516	-0.0219	0.083443	0.188832	0.00E+00
5	5	-0.14295	0.036055	-0.21658	-0.14295	-0.06933	6.52E-19

Result 5

res5$dic

$dic					
[1]	-29.1076				
$p.eff					
[1]	5.609777				
$local.dic					
[1]	-5.82148	-5.822	-5.82206	-5.82119	-5.8209
$local.p.eff					
[1]	1.122281	1.121476	1.12172	1.122045	1.122255
$mean.deviance					
[1]	-34.7174				
$deviance.mean					
[1]	-40.3272				

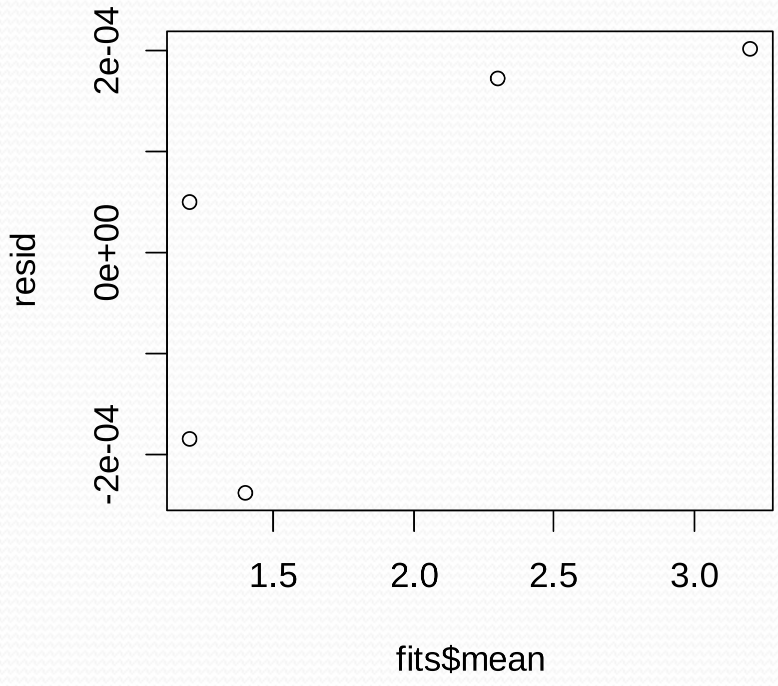
Result 6

fits<-res5$summary.fitted.values

	mean	sd	0.025quan	0.5quant	0.975quant
fitted.predictor.1	1.19995	0.009922	1.178116	1.19997	1.221611
fitted.predictor.2	1.400239	0.009922	1.37889	1.400141	1.422388
fitted.predictor.3	2.299828	0.009954	2.277815	2.299898	2.32127
fitted.predictor.4	3.199799	0.009961	3.177718	3.199881	3.221213
fitted.predictor.5	1.200185	0.009972	1.178718	1.200109	1.222271

Plot: residuals versus fitted values

plot(fits$mean,resid$mean)



Random effect example

```
formula3<-y~1+x1+f(ind,model='iid')
res3<-inla(formula3,family="gaussian",data=As,
control.compute=list(dic=TRUE))
```

Result 1

summary(res3)

Fixed	effects:					
	mean	sd	0.025quan	0.5quant	0.975quan	kld
(Intercept)	1.2379	0.9152	-0.6321	1.2378	3.1084	0
x1	0.1863	0.2491	-0.3228	0.1863	0.6955	0
Random	effects:					
Name	Model					
ind	IID	model				

Result 2

Deviance Information Criterion: -28.49			
Effective number of parameters: 6.498			
Marginal Likelihood: -19.74			

Results 3

res3$summary.fixed

	mean	sd	0.025quan	0.5quant	0.975quan	kld
(Intercept)	1.237863	0.915233	-0.63213	1.237842	3.108409	0
x1	0.186271	0.249148	-0.3228	0.186269	0.695464	0

res3$summary.random

$ind			
	ID	mean	sd
1	1	-0.2427	0.675712
2	2	-0.26622	0.460814
3	3	0.428789	0.381372
4	4	1.123752	0.478236
5	5	-1.04347	0.639186

Results 4

res3$summary.fitted.values

	mean	sd	0.025quan	0.5quant	0.975quan
fitted.predictor.1	1.200046	0.009898	1.17808	1.200026	1.222164
fitted.predictor.2	1.40005	0.009904	1.378076	1.400029	1.422191
fitted.predictor.3	2.299918	0.009948	2.277721	2.299953	2.321854
fitted.predictor.4	3.199786	0.009949	3.177373	3.199877	3.221502
fitted.predictor.5	1.200198	0.009938	1.178477	1.200114	1.222554