Session 5.2: Missing data inputation

Imperial College London

Learning Objectives

After this session you should be able to:

- Appreciate the importance of thinking about why data are missing, and stating your modelling assumptions
- Understand the disadvantages of complete case analysis
- Learn about how Bayesian methods can be used for handling missing data
- Be able to run the above models in R-INLA

The topics covered in this lecture are covered in Chapter 12 of Gómez-Rubio (2020):
https://becarioprecario.bitbucket.io/inla-gitbook/

Why we care about missing data?

- Missing data are common!
- Usually inadequately handled in both observational and experimental research

- For example, Wood, White, and Thompson (2004) reviewed 71 recently published BMJ, JAMA, Lancet and NEJM papers
  – 89% had partly missing outcome data
  – In 37 trials with repeated outcome measures, 46% performed complete case analysis
  – Only 21% reported sensitivity analysis

- Sterne, White, Carlin, Spratt, Royston, Kenward, Wood, and Carpenter (2009) reviewed articles using Multiple Imputation in BMJ, JAMA, Lancet and NEJM from 2002 to 2007
  – 59 articles found, with use doubling over 6 year period
  – However, the reporting was almost always inadequate

Outline

1. How do missing data arise?

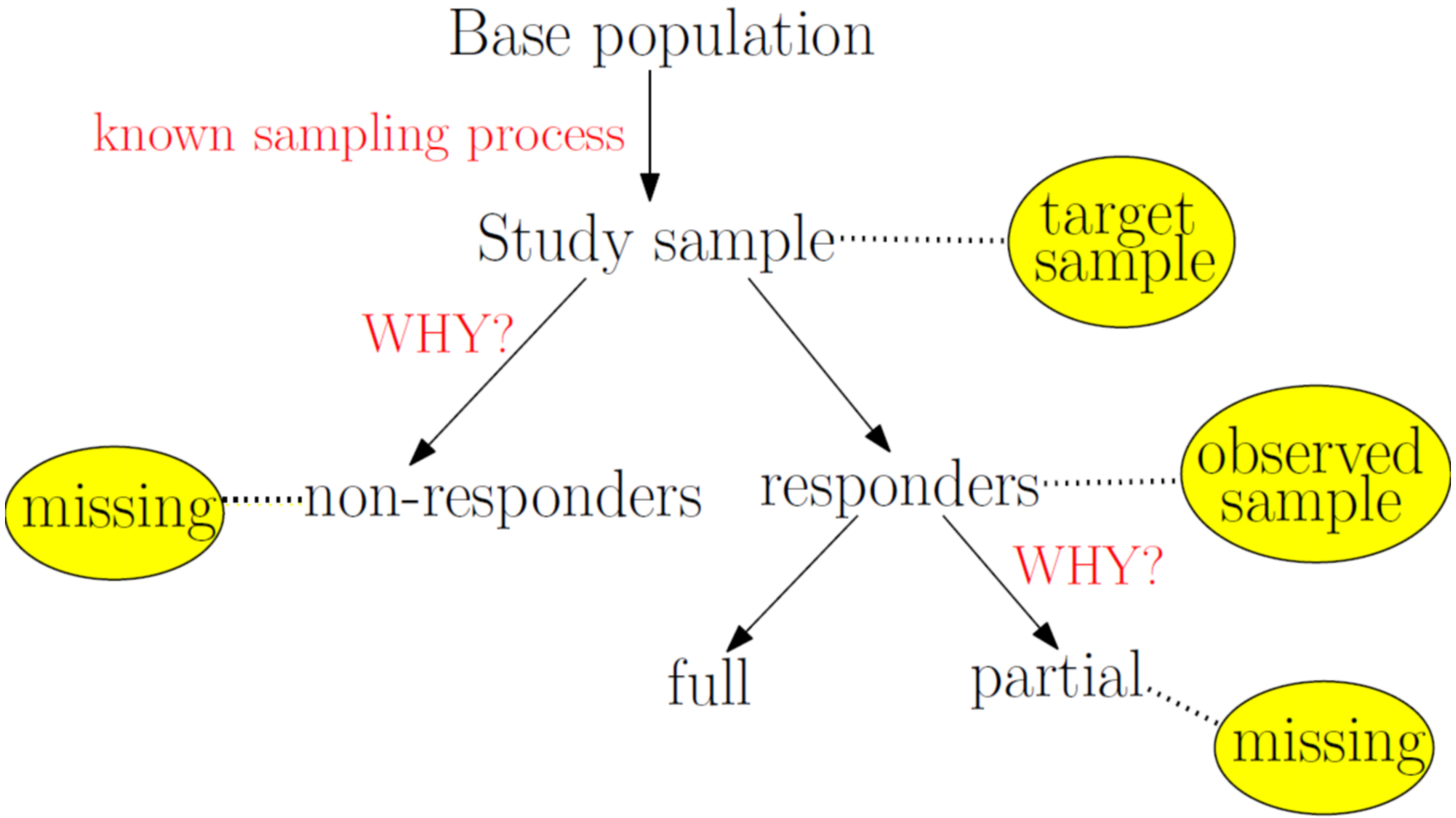
2. Example: children height and weight

3. Bayesian imputation

4. Extending the model

How do missing data arise?

How do missing data arise?



Different types of missing data: MCAR

- There are three types of missing data, depending on why the missingness arise. Let's define m_i as the variable indicating if the $i-th$ observation is missing

$$m_i \sim \text{Bernoulli}(p_i)$$

1. Missing completely at random (MCAR) occurs when the missing data are independent from the observed or unobserved data:

$$\text{logit}(p_i) = \theta_0$$

This means that the missing values can be ignored and the analysis can be conducted as usual.

Different types of missing data: MAR and MNAR

2. Missing at random (MAR) occurs when the missing data depends ONLY on the observed data:

$$\text{logit}(p_i) = \theta_0 + \mathbf{x_i}\theta_1$$

In this case, this can be introduced into the model so that missing observations are imputed as part of the model fitting.

Different types of missing data: MAR and MNAR

2. Missing at random (MAR) occurs when the missing data depends ONLY on the observed data:

$$\text{logit}(p_i) = \theta_0 + \mathbf{x_i}\theta_1$$

In this case, this can be introduced into the model so that missing observations are imputed as part of the model fitting.

3. Missing non at random (MNAR) occurs when the missing data depends on both the observed and missing data:

$$\text{logit}(p_i) = \theta_0 + \mathbf{x_i}\theta_1 + \lambda\mathbf{y_i}$$

This scenario is difficult to tackle since there is no information about the missingness mechanism and the missing data.

Missing response or missing covariates

- Additionally, it is crucial to distinguish between missing values in the response and in the covariates.

- When the missingness is in the response (and it is MCAR or MAR), these can naturally be predicted as the distribution of the response values is determined by the statistical model to be fit (posterior predictive distribution in the Bayesian approach)

- Missingness in the covariats requires additional steps:
  - A model needs to be specified on the covariate (imputation model) if the missingness mechanism is MCAR or MAR
  - An additional model of missingness needs to be specified if the mechanism is MNAR
  - Here we will consider only missing values in the response
  - `R-INLA` requires a certain degree of complexity to deal with missing values in covariates, if you are interested in learning more look at Gómez-Rubio, Cameletti, and Blangiardo (2022)

Example: height and weight of children

Example: height and weight of children

- Information of 10,030 children measured within the Fifth Dutch Growth Study 2009 (Schönbeck, Talma, Van Dommelen, Bakker, Buitendijk, HiraSing, and Van Buuren, 2013)
- Data available from the `fdgs` dataset in the `library(mice)` (Van Buuren and Groothuis-Oudshoorn, 2011)

Variable	Description	Missing
id	Child ID	0
reg	Region (5 levels)	0
age	Age (year)	0
sex	Sex	0
hgt	Height (cm)	23
wgt	Weight (kg)	20
hgt.z	Re-scaled height (as a Z-score)	23
wgt.z	Re-scaled weight (as a Z-score)	20

Summarising the data

- The data can be summarised using

```
> options(width = 100)
> summary(fdgs)
```

```
      id              reg             age               sex             hgt              wgt        
Min.   :100001   North: 732   Min.   : 0.008214   boy :4829   Min.   : 46.0   Min.   :  2.585  
1st Qu.:106353   East :2528   1st Qu.: 1.618754   girl:5201   1st Qu.: 83.8   1st Qu.: 11.600  
Median :203855   South:2931   Median : 8.084873               Median :131.5   Median : 27.500  
Mean   :180091   West :2578   Mean   : 8.157936               Mean   :123.9   Mean   : 32.385  
3rd Qu.:210591   City :1261   3rd Qu.:13.547570               3rd Qu.:162.3   3rd Qu.: 51.100  
Max.   :401955                Max.   :21.993155               Max.   :208.0   Max.   :135.300  
                                                              NA's   :23      NA's   :20       

     hgt.z              wgt.z        
Min.   :-4.470000   Min.   :-5.04000  
1st Qu.:-0.678000   1st Qu.:-0.62475  
Median :-0.019000   Median : 0.02600  
Mean   :-0.006054   Mean   : 0.04573  
3rd Qu.: 0.677000   3rd Qu.: 0.70700  
Max.   : 3.900000   Max.   : 4.74100  
NA's   :23          NA's   :20        
```

- Note that several variables in the dataset have missing observations. In particular, height (`hgt`) and weight (`wgt`), which are common variables used as response or predictors in models.

Subsetting the data

- In order to provide a smaller dataset to speed up computations, only the children with missing values (in height and weight) and another 1000 ones taken at random will be used in the analysis

```
> # Subsect 1, observations with NA's
> subset1 <- which(is.na(fdgs$wgt) | is.na(fdgs$hgt))
>
> #Subset 2, random sample of 1000 individuals
> set.seed(1)
> subset2 <- sample((1:nrow(fdgs))[-subset1], 1000)
>
> # Subset 1 + subset 2
> fdgs.sub <- fdgs[c(subset1, subset2), ]
```

Sumamry of the data

```
> summary(fdgs.sub)
```

```
      id              reg           age              sex          hgt              wgt        
Min.   :100098   North: 78   Min.   : 0.07118   boy :493   Min.   : 46.00   Min.   :  2.585  
1st Qu.:106293   East :275   1st Qu.: 1.74264   girl:550   1st Qu.: 86.28   1st Qu.: 11.960  
Median :204306   South:298   Median : 8.59411              Median :136.05   Median : 29.000  
Mean   :183214   West :250   Mean   : 8.56536              Mean   :127.04   Mean   : 33.614  
3rd Qu.:211388   City :142   3rd Qu.:14.16427              3rd Qu.:165.10   3rd Qu.: 53.100  
Max.   :401949               Max.   :21.88364              Max.   :199.00   Max.   :117.300  
                                                           NA's   :23       NA's   :20       

     hgt.z              wgt.z        
Min.   :-4.26300   Min.   :-4.0750  
1st Qu.:-0.66800   1st Qu.:-0.6520  
Median :-0.04550   Median :-0.0070  
Mean   :-0.02313   Mean   : 0.0202  
3rd Qu.: 0.64550   3rd Qu.: 0.6960  
Max.   : 3.20000   Max.   : 3.6300  
NA's   :23         NA's   :20       
```

Bayesian imputation

Model specification

- We first predict weight as a function of age and sex. We specify:

$$wgt_i \sim \text{Normal}(\alpha + \beta_1 sex_i + \beta_2 age_i, \sigma^2)$$

- where `wgt` is missing we will have `NA` in the corresponding vector, e.g.

```
> fdgs.sub$wgt[1:10]
```

```
[1] 23.200     NA     NA     NA  8.445     NA  6.960 10.420     NA 14.100
```

- Remember the posterior predictive distribution (presented in lecture 2.2):

$$p(\mathbf{y}^*|\mathbf{y}) = \int p(y^*|\theta)p(\theta|\mathbf{y})d\theta$$

  – here $\mathbf{y}^*$ identifies the missing values, while $\mathbf{y}$ is the set of observed values for `wgt`

Running the model in R-INLA

```
> library("INLA")
> wgt.inla <- inla(wgt ~ age + sex, data = fdgs.sub,
+ control.predictor = list(compute = TRUE), control.compute=list(return.marginals.predictor=TRUE))
> wgt.inla$summary.fixed
```

```
                mean         sd 0.025quant  0.5quant 0.975quant      mode          kld
(Intercept)  6.259187 0.43617515   5.403716  6.259187   7.114657  6.259187 8.506052e-12
age          3.330942 0.03370539   3.264835  3.330942   3.397048  3.330942 8.599942e-12
sexgirl     -2.378944 0.44629314  -3.254258 -2.378944  -1.503627 -2.378944 8.499316e-12
```

```
> wgt.inla$summary.hyperpar
```

```
                                                    mean           sd 0.025quant   0.5quant 0.975quant
Precision for the Gaussian observations 0.01972451 0.0008707156  0.0180516 0.01971203 0.02147299
                                                    mode
Precision for the Gaussian observations 0.01968792
```

- Note that we need to include `control.predictor=list(compute = TRUE)` so `INLA` can estimate the predictive distribution for the missing observations, while `control.compute=list(return.marginals.predictor=TRUE)` tells inla that we want to access the entire posterior distribution of the prediction (rather than only the summary)

Getting the posterior prediction

We now subset the children indexes with missing values so we can report their predictive distributions:

```
> wgt.na = which(is.na(fdgs.sub$wgt))
> rownames(fdgs.sub)[wgt.na]
```

```
 [1] "275"  "1278" "1419" "2135" "2684" "2940" "3069" "3189" "3543" "4262" "5687" "6485" "7101"
[14] "7108" "7506" "8064" "8065" "8067" "8098" "8588"
```

```
> # Obtain the predictive distribution
> wgt.inla$summary.fitted.values[wgt.na, c("mean", "sd")][1:5,]
```

```
                           mean        sd
fitted.Predictor.0002 23.067927 0.3203228
fitted.Predictor.0003 27.617341 0.3331316
fitted.Predictor.0004 54.019923 0.3768173
fitted.Predictor.0006 13.144501 0.3930171
fitted.Predictor.0009  7.838159 0.3939883
```

Remember that you can also access the marginal posterior distributions (rather than the summary) using `wgt.inla$marginals.fitted.values`

Imputing hgt using the same approach

- Similarly, a model can be fit to explain height based on age and sex and to compute the predictive distribution of the missing observations:

```
> hgt.inla = inla(hgt ~ age + sex, data = fdgs.sub,
+ control.predictor = list(compute = TRUE),
+ control.compute = list(return.marginals.predictor=TRUE))
> hgt.inla$summary.fixed
```

```
                mean         sd 0.025quant  0.5quant 0.975quant      mode          kld
(Intercept) 77.148115 0.72150119  75.733029 77.148117  78.563195 77.148117 7.425339e-12
age          6.022575 0.05552369   5.913676  6.022575   6.131473  6.022575 8.643032e-12
sexgirl     -4.716066 0.73157410  -6.150898 -4.716068  -3.281220 -4.716068 8.495129e-12
```

```
> hgt.inla$summary.hyperpar
```

```
                                                    mean           sd  0.025quant    0.5quant
Precision for the Gaussian observations 0.007395396 0.0003241965 0.006762723 0.007391652
                                                0.975quant        mode
Precision for the Gaussian observations 0.008044646 0.007383216
```

Imputing hgt using the same approach

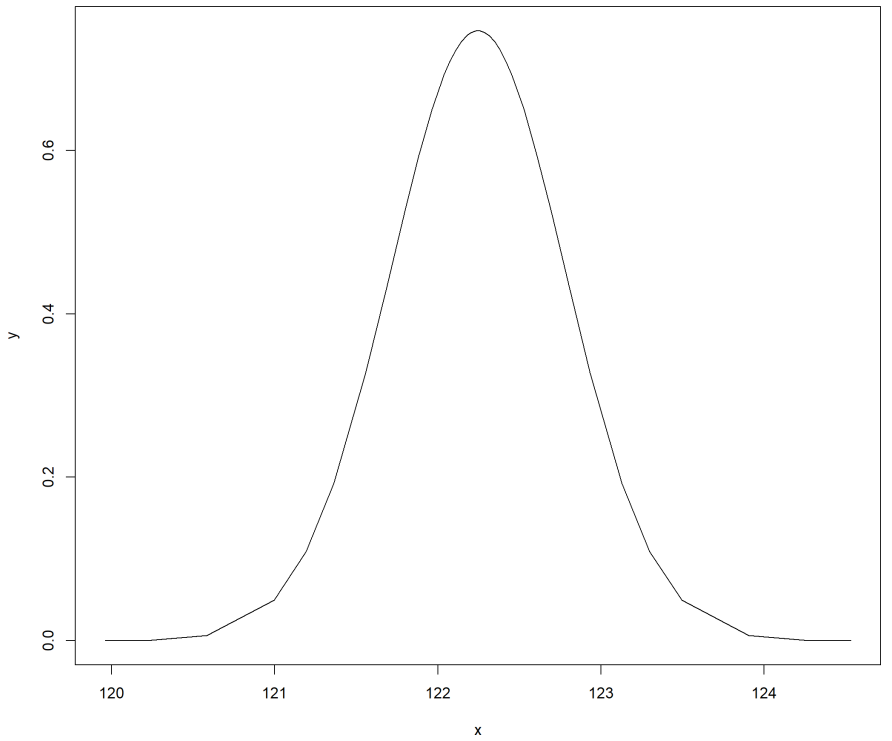
- We can obtain the predictions using

```
> hgt.na = which(is.na(fdgs.sub$hgt))
> hgt.inla$summary.fitted.values[hgt.na, c("mean'
```

	mean	sd
fitted.Predictor.0001	122.24529	0.5364459
fitted.Predictor.0005	83.24901	0.6844784
fitted.Predictor.0007	74.64156	0.6798160
fitted.Predictor.0008	82.80381	0.6870701
fitted.Predictor.0010	88.26140	0.6024006
fitted.Predictor.0012	96.55556	0.6146600
fitted.Predictor.0013	86.31595	0.6670367
fitted.Predictor.0016	81.84746	0.6926869
fitted.Predictor.0017	77.75821	0.7174290
fitted.Predictor.0018	81.59988	0.6383572

- We can plot the entire posterior distributions for each child with:

```
> # First child with missing value
> plot(hgt.inla$marginals.fitted.values[[hgt.na[1
```



Extending the model

Joint model of height and weight

- The two previous models consider height and weight separately, but it is clear that there is a high correlation between height and weight, which is caused by the age and sex of the child.

```
[1] 0.9999865
```

- We can build a joint model for height and weight to exploit a correlated effect between the coefficients of age in both models.

$$hgt_i = \alpha_h + \beta_{h1} sex_i + \beta_{h2} age_i + \epsilon_{1i}$$
$$wgt_i = \alpha_w + \beta_{w1} sex_i + \beta_{w2} age_i + \epsilon_{2i}$$

with

- α_h, α_w model intercepts
- β_{h1}, β_{w1} the effect of sex
- β_{h2}, β_{w2} the effect of age
- ϵ_1, ϵ_2 the error terms (note that this specification is equivalent to the one above for the separate models)

Joint model of height and weight: prior

- The vectors (β_{1h}, β_{1w}) and (β_{2h}, β_{2w}) are modeled using a multivariate Gaussian distribution with mean 0 and covariance matrix with $1/\tau_{hj}$ and $1/\tau_{wj}$ as variances, and $\rho_j/\sqrt{(\tau_{jh}\tau_{jw})}$ as the covariance where ρ_j is the correlation parameter.
- First, the bivariate response variable needs to be put in a two-column matrix given that the model will be made of two data distributions

```
> n = nrow(fdgs.sub)
> y = matrix(NA, nrow = 2 * n, ncol = 2)
> y[1:n, 1] = fdgs.sub$hgt
> y[n + 1:n, 2] = fdgs.sub$wgt
```

- Similarly, as we have two intercepts, we need to define these explicitly as covariates with all values equal to one:

```
> I = matrix(NA, nrow = 2 * n, ncol = 2)
> I[1:n, 1] = 1
> I[n + 1:n, 2] = 1
```

Correlated effects

- Now we need to define the correlated effects. We will use the random effect specification similar to what we saw with the hierarchical models (`iid`), but modified to have the two coefficients as correlated `f(...,model=iid2d)`.
- In order to do so we need to modify the variables `age` and `sex` as they will be passed to the model as weights of the latent random effects `iid2d` specification. We need them to be twice as long to match the dimension of the response:

```
> age.joint = rep(fdgs.sub$age, 2)
> sex.joint = rep(fdgs.sub$sex, 2)
```

- Finally we need two index vectors to indicate which coefficient to use from the `iid2d` model is required. These indexes will be 1 for the first half of observations (to indicate that the coefficient is β_h) and 2 for the second half (to indicate that the coefficient is β_w).

```
> idx.age = rep(1:2, each = n)
> idx.sex = rep(1:2, each = n)
```

Model fitting

The model is fit and summarized as seen below

```
> # Model formula
> joint.f = y ~ -1 + I + f(idx.sex, sex, model="iid2d", n=2) + f(idx.age, age, model = "iid2d", n = 2)
> # Model fit
> fdgs.joint = inla(joint.f,
+   data = list(y = y, I = I, sex = sex.joint, age = age.joint, idx.age = idx.age, idx.sex = idx.sex),
+   family = rep("gaussian", 2),
+   control.predictor = list(compute = TRUE))
> # Summary fixed (intercept)
> fdgs.joint$summary.fixed
```

```
       mean        sd 0.025quant  0.5quant 0.975quant      mode          kld
I1 80.39418 1.1103714  78.230017 80.389852  82.582920 80.389796 4.452303e-09
I2  8.58588 0.6944199   7.232892  8.582641   9.957273  8.582617 5.505778e-09
```

Hyperparameters

- The hyperparameters can be obtained through

```
> fdgs.joint$summary.hyperpar[,1:5]
```

```
                                                    mean            sd  0.025quant    0.5quant
Precision for the Gaussian observations    0.007384289 0.0003034016 0.006816054 0.007374328
Precision for the Gaussian observations[2] 0.019764793 0.0008283452 0.018189014 0.019745778
Precision for idx.sex (component 1)        0.470089210 0.0442688411 0.395293604 0.466514161
Precision for idx.sex (component 2)        0.881830101 0.1135674860 0.639700696 0.886038462
Rho1:2 for idx.sex                         0.854849683 0.0114490963 0.835076086 0.854691634
Precision for idx.age (component 1)        0.110412222 0.0085333429 0.091359517 0.110446031
Precision for idx.age (component 2)        0.318005972 0.0377428687 0.235637882 0.320288517
Rho1:2 for idx.age                         0.968896310 0.0025834910 0.963975317 0.968965331
                                            0.975quant
Precision for the Gaussian observations    0.008010606
Precision for the Gaussian observations[2] 0.021450099
Precision for idx.sex (component 1)        0.569357127
Precision for idx.sex (component 2)        1.070437584
Rho1:2 for idx.sex                         0.879601588
Precision for idx.age (component 1)        0.123905000
Precision for idx.age (component 2)        0.377084846
Rho1:2 for idx.age                         0.974250923
```

Variable effects

- While the coefficients for age and sex are part of the random effects of the model:

```
> #Sex
> fdgs.joint$summary.random$idx.sex
```

```
  ID      mean        sd 0.025quant  0.5quant 0.975quant      mode          kld
1  1 -3.763250 0.6062827  -4.962474 -3.759162  -2.587257 -3.759101 1.433920e-08
2  2 -2.347257 0.3854653  -3.112132 -2.344237  -1.599495 -2.344202 1.522774e-08
```

```
> #Age
> fdgs.joint$summary.random$idx.age
```

```
  ID     mean         sd 0.025quant 0.5quant 0.975quant     mode          kld
1  1 6.022913 0.05536535   5.914333 6.022912   6.131494 6.022912 5.795005e-13
2  2 3.331332 0.03362666   3.265391 3.331330   3.397287 3.331330 2.356426e-12
```

Estimates of missing values

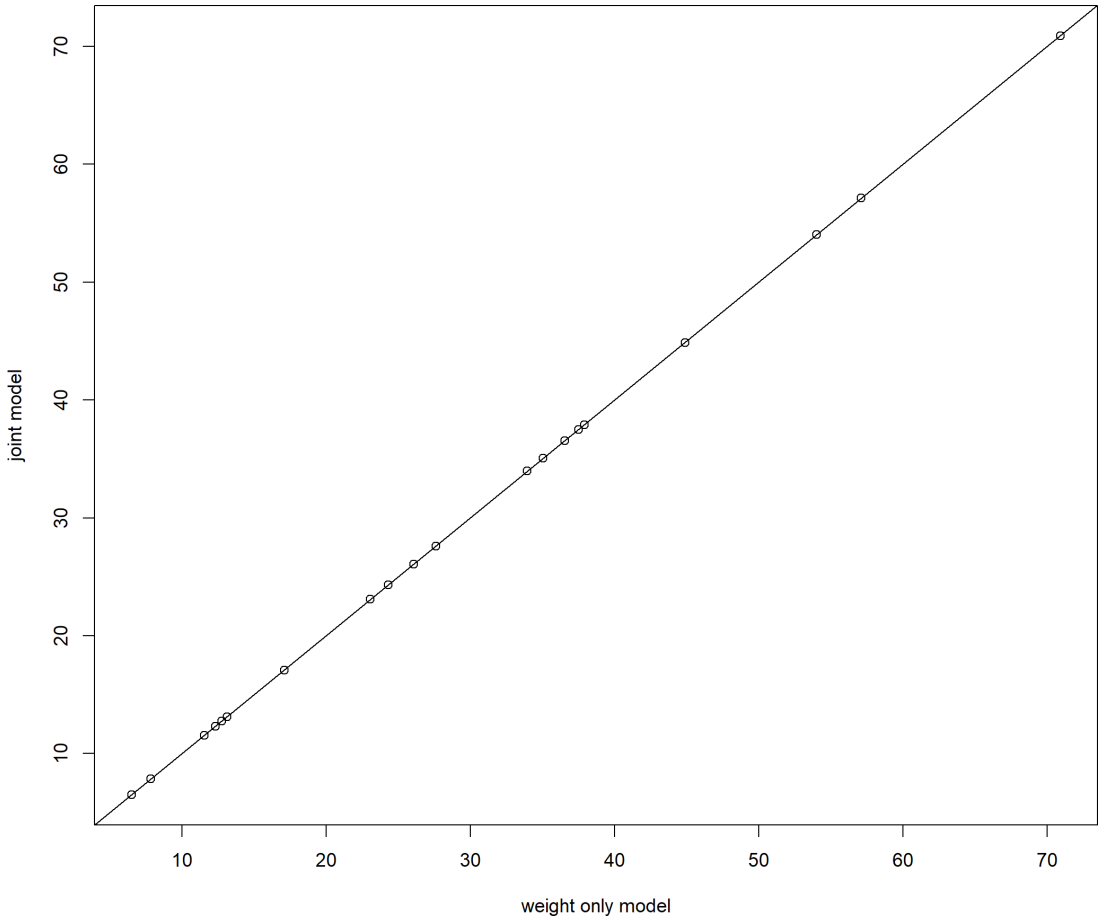
- Finally to access the predicted values for the children with missing data we need to remember that we now have the response `y` stacked (first height, then weight)
- As height is the first variable we can use `hgt.na` to get to the indexes of the children with missing data
- For weight we need to use `wgt.na` and add `n` to each index, as this is the total number of observations, so the data for weight will start on the index 1044

```
> joint.wgt.na = wgt.na+n
```

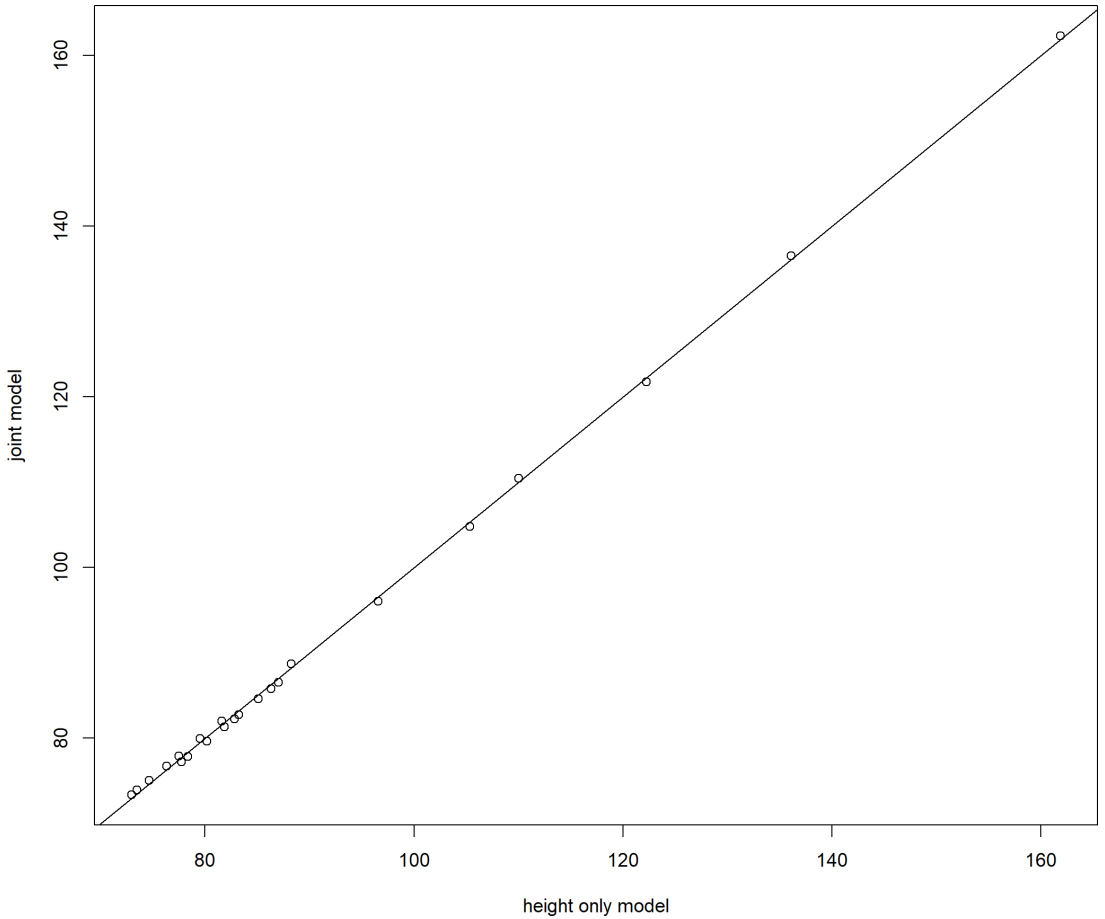
```
> #Height
> fdgs.joint$summary.fitted.values[hgt.na, c("mean", "sd")][1:10,]
> #Weight
> fdgs.joint$summary.fitted.values[joint.wgt.na, c("mean", "sd")][1:10,]
```

Comparing the results of the joint and separate models

Weight



Height



References

Gómez-Rubio, V. (2020). *Bayesian inference with INLA*. CRC Press.

Gómez-Rubio, V., M. Cameletti, and M. Blangiardo (2022). "Missing data analysis and imputation via latent Gaussian Markov random fields". In: *SORT-Statistics and Operations Research Transactions*, pp. 217-244.

Schönbeck, Y., H. Talma, P. Van Dommelen, et al. (2013). "The world's tallest nation has stopped growing taller: the height of Dutch children from 1955 to 2009". In: *Pediatric Research* 73.3, pp. 371-377.

Sterne, J. A., I. R. White, J. B. Carlin, et al. (2009). "Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls". In: *BMJ* 338.

Van Buuren, S. and K. Groothuis-Oudshoorn (2011). "mice: Multivariate imputation by chained equations in R". In: *Journal of Statistical Software* 45, pp. 1-67.

Wood, A. M., I. R. White, and S. G. Thompson (2004). "Are missing outcome data adequately handled? A review of published randomized controlled trials in major medical journals". In: *Clinical trials* 1.4, pp. 368-376.