

# Modeling the Effects of Nutrition with Mixed-Effect Bayesian Network

*Jari Turkia, jari.turkia@cgi.com*  
*CGI, University of Eastern Finland*

## Abstract

This work proposes Mixed-Effect Bayesian Network (MEBN) as a method for modeling the effects of nutrition. It allows to identify both typical and personal correlations between nutrients and their bodily responses. Predicting a personal network of nutritional reactions would allow interesting applications at personal diets and in understanding this complex system. Brief theory of MEBN is given followed by implementation in R and Stan. A real life dataset from a nutritional Sysdimet study is then analyzed with the method and the results are visualized with an interactive JavaScript-visualization.

## The Effects of Nutrition

Nutritional therapists have known for a long that people can react differently to the same nutrition. I am proposing a Mixed-Effect Bayesian Network (MEBN) as a method for modeling the effects of nutrition in both population and personal level. By the effects of nutrition we mean the way how people react to different levels of nutrients at their diets. There have been studies of these effects on limited cases, like personal glucose metabolism (Zeevi et al. 2015), but MEBN would allow more general modeling.

The Bayesian Networks (BN) are directed acyclic graphical models that contain both observed and latent random variables as vertices and edges as indicators of connection. In the setting of nutritional modeling the observed variables are nutrients at person's diet and their corresponding bodily responses, like blood characteristics. The connections between these variables can be very complex and the BNs seem to be intuitively appealing method for modeling such a system.

Besides of the typical correlations at the system, we are interested in the personal variations at the reaction types. Are some people more sensitive to some nutrient than the others? If this information could be systemically quantified, it would allow us to predict personal networks of reaction types. This in turn, opens up lots of new applications in personal recommendations and health care.

For capturing these personal variances we need a dataset that contains several repeated measurements from number of persons. Measurements from each person are correlated with each other and a general method for modeling this correlation is hierarchical, or mixed-effect, model. The same model can be used to estimate the reaction type in both population and personal level.

Previously BNs have been considered mainly for uncorrelated observations (Scutari and Denis 2014, Nagarajan and Scutari (2013), Aussem et al. (2010)), but in this work we are using Mixed-Effect Bayesian Network parametrization (Bae et al. 2016) that allows combining the hierarchical mixed-effect modeling and Bayesian Network as a general framework for expressing the system.

This presentation covers first briefly the theory of graphical models and how it can be expanded to correlated observations. Then it is shown how this modeling can be implemented with Stan and what benefits fully Bayesian estimation can offer in understanding the uncertainty at the model.

## Mixed-Effects Bayesian Network

Let us denote the graph of interconnected nutrients and responses with  $G$ . We can then formulate the modeling problem as finding the graph  $G$  that is most probable given the data  $D$

$$P(G|D) \tag{1}$$

By using Bayes' Rule we can be split this probability into proportions of data likelihood with given graph and any prior information we might have about suitable graphs

$$P(G|D) \propto P(D|G)P(G) \tag{2}$$

Now the problem is converted into a search of the maximum likelihood graph for given data. If all the graphs are equally probable then  $P(G)$  is a constant and does not affect the search, but it can be also beneficial to use it to guide the search towards meaningful graphs (Bishop:2006:PRM:1162264, Nagarajan and Scutari 2013).

**Decomposition of the likelihood.** Bayesian network factorizes into local distributions according to *local Markov property* stating that a variable  $X_i$  is independent of its non-descendants given its parents at the graph. The *global Markov property* states that a variable is independent of all the remaining variables in the graph conditionally on its *Markov blanket* that consists its parents and child nodes at the graph, and additional parents of the child nodes (Bae et al. 2016, Koller and Friedman (2009)). With this decomposition the joint probability of the graph can be calculated with sum and product rules of probability as a product of the independent local graphs  $G_i$ . The graph structure depends also on the parameters  $\phi_i$  describing the dependencies and they should be taken into account at the estimation. We assume that  $\phi_i$  is a set that pools all the parameters that describe the relationship.

Likelihood of the data is then

$$P(D|G) = \prod_{i=1}^v P(X_i|pa(X_i), \phi_i, G_i)P(\phi_i|G_i) \tag{3}$$

assuming we have  $v$  independent local distributions at the graph. The notation  $pa(X_i)$  denotes the parent variables of variable  $X_i$  according to graph structure  $G_i$ . Since the probability of data in the graph depends on the parameters of the local distributions,  $\phi_i$ , they have to be integrated out from the equation to make the probability of graph independent of any specific choice of parameters

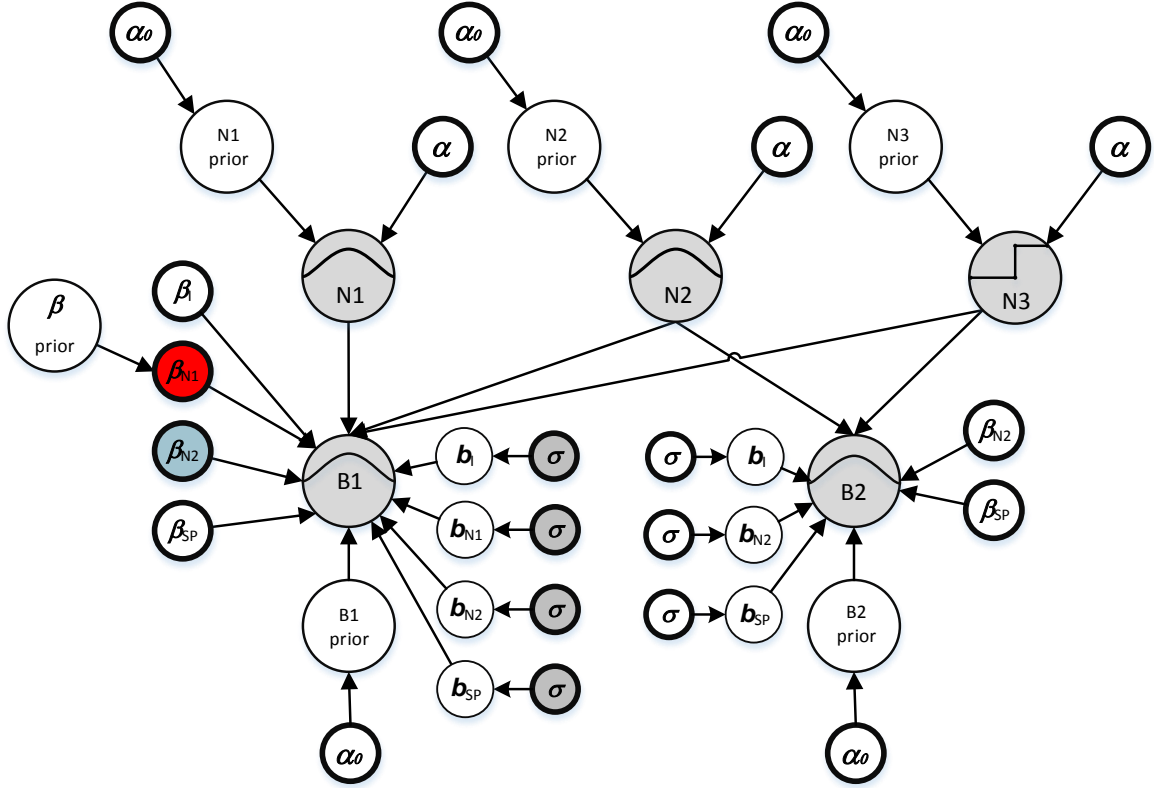
$$\prod_{i=1}^v \int P(X_i|pa(X_i), \phi_i, G_i)P(\phi_i|G_i)d\phi_i = \prod_{i=1}^v P(X_i|pa(X_i), G_i)P(G_i) = P(D|G) \tag{4}$$

Besides the Markov properties, we also assume *global independence of the parameters*

$$\phi_i \neq \phi_j, i \neq j \tag{5}$$

and for Bayesian estimation we assume *hyper-Markov law* (Dawid and Lauritzen 1993) for ensure that these decompositions are indeed independent.

**Linear dependency between variables.** As we are more interested in the system and less considered about details of any specific nutritional response, we consider it adequate to model the dependency between the nutrients and bodily responses with an approximate linear model. However, simple linear model is not enough, but we need a parametrization that is able to reflect the correlations between observations and express the amount of variability between persons since the data consists of several repeated measurements from different persons.



Generally, the local probability distributions can be from exponential family of distributions, but in this example we consider only normally distributed response variables. Subset of parent nodes possibly containing personal variance is denoted with  $pa_Z(X_i)$ . For mixed-effect modeling we assume parameters  $\phi_i = \{\beta_i, b_i\}$  for expressing typical and personal reaction types. In multivariate normal model the uncertainty is furthermore defined by variance-covariance matrix  $V_i$

$$P(X_i|pa(X_i), \phi_i) = N(X_i|pa(X_i)\beta_i + pa_Z(X_i)b_i, V_i) \quad (6)$$

This theory motivates our search for optimal graph with Stan. By decomposing the joint likelihood into local probability distributions according to Markov properties it is possible to find the optimal graph by estimating one local distributions one by one.

## Estimating the Hierarchical Local Distributions with Stan

In the mixed-effect modeling the purpose is to explain some of the model's variance in  $V_i$  with the latent personal effect variables  $b_i$ . These in turn offer us a way to detect and express the personal variations in nutritional effects. Let us assume that matrix  $Z$  is a design matrix of personal effects. Then variance-covariance matrix is defined by

$$V = ZDZ' + R \quad (7)$$

where  $R$  is a variance-covariance matrix of residuals and  $D$  is a variance-covariance matrix of personal, or random-effects,

$$D = \mathcal{T}C\mathcal{T}' \quad (8)$$

where  $\mathcal{T}$  is a diagonal matrix of personal effect variances and  $C$  is correlation matrix that can be divided into Cholesky decompositions as

$$C = LL' \quad (9)$$

and with  $L$  we can define the personal effects as

$$b = Lu, u \sim N(0, I) \quad (10)$$

as we assume for now that personal random-effects are drawn from Normal distribution.

This is implemented in Stan as follows

```
transformed parameters {
  // ...
  // Create diagonal matrix from sigma_b and premultiply it with L
  D = diag_pre_multiply(sigma_b, L);

  // Group-level effects are generated by multiplying D with z that has standard normal distribution
  for(j in 1:J)
    b[j] = D * z[j];
}
```

The actual model with Normal distribution having the linear mixed-effect likelihood is defined below. Notice that instead of matrix  $V$ , in Stan we are using vectors `group` and scalar `sigma_e`.

```
model {
  // ...
  // Standard normal prior for random effects
  for (j in 1:J)
    z[j] ~ normal(0,1);

  // Likelihood
  // - link function (identity function for Normal dist.) for typical correlation
  mu = temp_Intercept + Xc * beta;

  // - add personal (group) effects
  for (i in 1:N)
  {
    mu[i] = mu[i] + Z[i] * b[group[i]];
  }

  // Y and mu are vectors, sigma_e is a scalar that is estimated for whole vector
  Y ~ normal(mu, sigma_e);
}
```

## Constructing the Population Level Graph of Nutritional Effects

The dataset in this example comes from Sysdimet study (Lankinen et al. 2011) that studied altogether 106 men and women with impaired glucose metabolism. For each person we have four observations on diet and

on blood tests. The blood tests are taken a week after the diet observation. We have picked few interesting variables indicating person’s diet, blood test results and personal information, like gender and medication.

There exist plenty of general algorithms for constructing BNs, but for this special case we can constrain the search to biologically plausible reaction graphs. We assume that all possible graphs are directed bipartite graphs with nutrients and personal information as root nodes and blood tests as targets.

```
# Read the data description
datadesc <- mebn.load_datadesc("Data description.xlsx")

# Read the actual data matching the description
sysdimet <- read.csv(file="data\\SYSDIMET_diet.csv", sep=";", dec=",")

# Define how to iterate through the graph
assumedpredictors <- datadesc[datadesc$Order==100,]
assumedtargets <- datadesc[datadesc$Order==200,]
```

**Pruning the edges.** The construction of the graph starts from fully connected graph, but for gaining nutritional knowledge of significant connections and also for efficient factorization of the BN likelihood, it is necessary to prune out the insignificant connections at the graph. For this we use shrinkage prior on beta coefficients to push the insignificant coefficients towards zero. Especially, we use regularized horseshoe prior (Piironen and Vehtari 2017a) that allows specifying the number of non-zero coefficients for each target. In the nutritional setting this provides a way for specifying prior knowledge about the relevant nutrients for each response. For now, we approximate that one third of the predictive nutrients are relevant, but finer approximation will be done based on previous nutritional research.

```
shrinkage_parameters <- within(list(),
{
  scale_icept <- 1          # prior std for the intercept
  scale_global <- 0.01821   # scale for the half-t prior for tau:
                           # (p0=6) / (D=22-6)*sqrt(n=106*4)

  nu_global <- 1           # degrees of freedom for the half-t priors for tau
  nu_local <- 1            # degrees of freedom for the half-t priors for lambdas
  slab_scale <- 1          # slab scale for the regularized horseshoe
  slab_df <- 1             # slab degrees of freedom for the regularized horseshoe
})
```

If the shrinkage prior does not shrink the coefficients to exactly zero, we are pruning out the insignificant connections with following test. Notice that in the population level graph we are keeping the connections that have large variance between persons even though they are not typically relevant. Personal variance means that the connection is relevant for someone.

The effect of shrinkage can be studied by using an alternative with Stan model “BLMM.stan” that omits the shrinkage.

```
my.RanefTest <- function(localsummary, PredictorId)
{
  abs(localsummary$fixef[PredictorId]) > 0.001 ||
  localsummary$ranef_sd[PredictorId] > 0.05
}
```

To assure that this pruning does not affect predicting accuracy of the model a projection method (Piironen and Vehtari 2017b) could be also used here. In projection approach the edges are removed if it does not affect the distance from the true model measured with KL-divergence.

**Construction of the graph.** The data structure of the graph is based on iGraph package. The process of BN construction starts by adding a node for every observed variable at the dataset.

```
# Add data columns describing random variables as nodes to the graph
# - initial_graph is iGraph object with only nodes and no edges
initial_graph <- mebn.new_graph_with_randomvariables(datadesc)
```

The joint distribution of the Bayesian Network is decomposed as local distributions, one for each target variable. These local distributions correspond to hierarchical regression models that are estimated with Stan and the resulting MCMC sampling is cached to files.

For estimating the typical graph, we iterate the hierarchical Stan-model through all the assumed predictors and targets. The result is an iGraph object that contains a directed bipartite graph with relating nutrients as predictors and blood test results as targets. We normalize all the values to unit scale and centering before the estimation. Instead of sampling, it is also possible to estimate the same model with Stan implementation of variational Bayes by switching the local\_estimation-paramter.

```
sysdimet_graph <- mebn.typical_graph(reaction_graph = initial_graph,
                                     inputdata = sysdimet,
                                     predictor_columns = assumedpredictors,
                                     assumed_targets = assumedtargets,
                                     group_column = "SUBJECT_ID",
                                     local_estimation = mebn.sampling,
                                     local_model_cache = "models",
                                     stan_model_file = "mebn\\BLMM_rhs.stan",
                                     edge_significance_test = my.Raneftest,
                                     normalize_values = TRUE,
                                     reg_params = shrinkage_parameters)
```

```
## [1] "Loading models\\fshdl_blmm.rds"
## [1] "Loading models\\fsldl_blmm.rds"
## [1] "Loading models\\fsins_blmm.rds"
## [1] "Loading models\\fskol_blmm.rds"
## [1] "Loading models\\fpgluk_blmm.rds"
```

Sampling may still cause some divergent transitions and errors on estimating correlation matrix that is quite restricted data type. Estimations of parameters seem nevertheless realistic.

Now “sysdimet\_graph” is a Mixed Effect Bayesian Network for whole sample population. We can store it in GEXF-format and load to visualization for closer inspection. The visual nature of the Bayesian Networks, and the graphical models in general, provide a useful framework for creating visualizations. Comparing to schematic figure 1, the coefficients and latent variables are removed and denoted by different colors. Blue edge denotes typically negative correlation, red edge denotes positive correlation and gray shade denotes the amount of personal variation at the correlation.

It can be seen from the visualization that many of the nutrients affect to several responses. This multiresponse effect is not yet implemented to the model, but should be accounted. It allows, for example, to gain knowledge about responses that have missing measurements for some persons, but have known connections to other responses and predictors.

## Inference

The graph gives an overview and we can query some of the most interesting reactions. We can, for example, investigate most significant typical reactions by quering largest beta coefficients

```
allnodes <- V(sysdimet_graph)
beta <- allnodes[allnodes$type=="beta"]

typical_effects<-data.frame(matrix(NA, nrow=length(beta), ncol=0))
```

```

typical_effects$name <- beta$name
typical_effects$value <- beta$value

largest_typical_negative <- typical_effects[order(typical_effects$value),]
largest_typical_positive <- typical_effects[order(-typical_effects$value),]

```

### Largest typical negative effects

```

##              name          value
## 24 beta_kolestrolilaakitys_fsldl -0.026897912
## 68 beta_kolestrolilaakitys_fskol -0.011823168
## 3      beta_energia_fshdl -0.005528406
## 12     beta_hhydr_fshdl -0.005381225
## 45     beta_sukupuoli_fsins -0.004419155

```

### Largest typical positive effects

```

##              name          value
## 1      beta_sukupuoli_fshdl 0.017062310
## 46 beta_kolestrolilaakitys_fsins 0.008108980
## 51     beta_mufa_fsins 0.007027948
## 49     beta_rasva_fsins 0.005167793
## 50     beta_safa_fsins 0.003460460

```

It can be inspected that females have typically higher levels of HDL cholesterol and cholesterol medication, besides of lowering the cholesterol levels, also raises blood insuling level.

What we are really interested in, though, are the variations between persons. For this, we can query the largest variances of random-effects..

```

b_sigma <- allnodes[allnodes$type=="b_sigma"]

personal_variances<-data.frame(matrix(NA, nrow=length(b_sigma), ncol=0))
personal_variances$name <- b_sigma$name
personal_variances$value <- b_sigma$value

largest_personal_variance <- personal_variances[order(-personal_variances$value),]
head(largest_personal_variance, 10)

```

```

##              name          value
## 47     b_sigma_energia_fsins 0.15997602
## 3      b_sigma_energia_fshdl 0.10965785
## 25     b_sigma_energia_fsldl 0.10701078
## 48     b_sigma_prot_fsins 0.10266477
## 52     b_sigma_pufa_fsins 0.10041444
## 69     b_sigma_energia_fskol 0.10014573
## 50     b_sigma_safa_fsins 0.09951722
## 46 b_sigma_kolestrolilaakitys_fsins 0.09942475
## 57     b_sigma_sakkar_fsins 0.09720251
## 49     b_sigma_rasva_fsins 0.09665083

```

So, there seems to be large variance in how the energy intake affects to insulin and cholesterol levels.

Relevance of these estimations can be inspected from their posterior distributions.

```

library("bayesplot")

# Local distribution for fsins

```

```

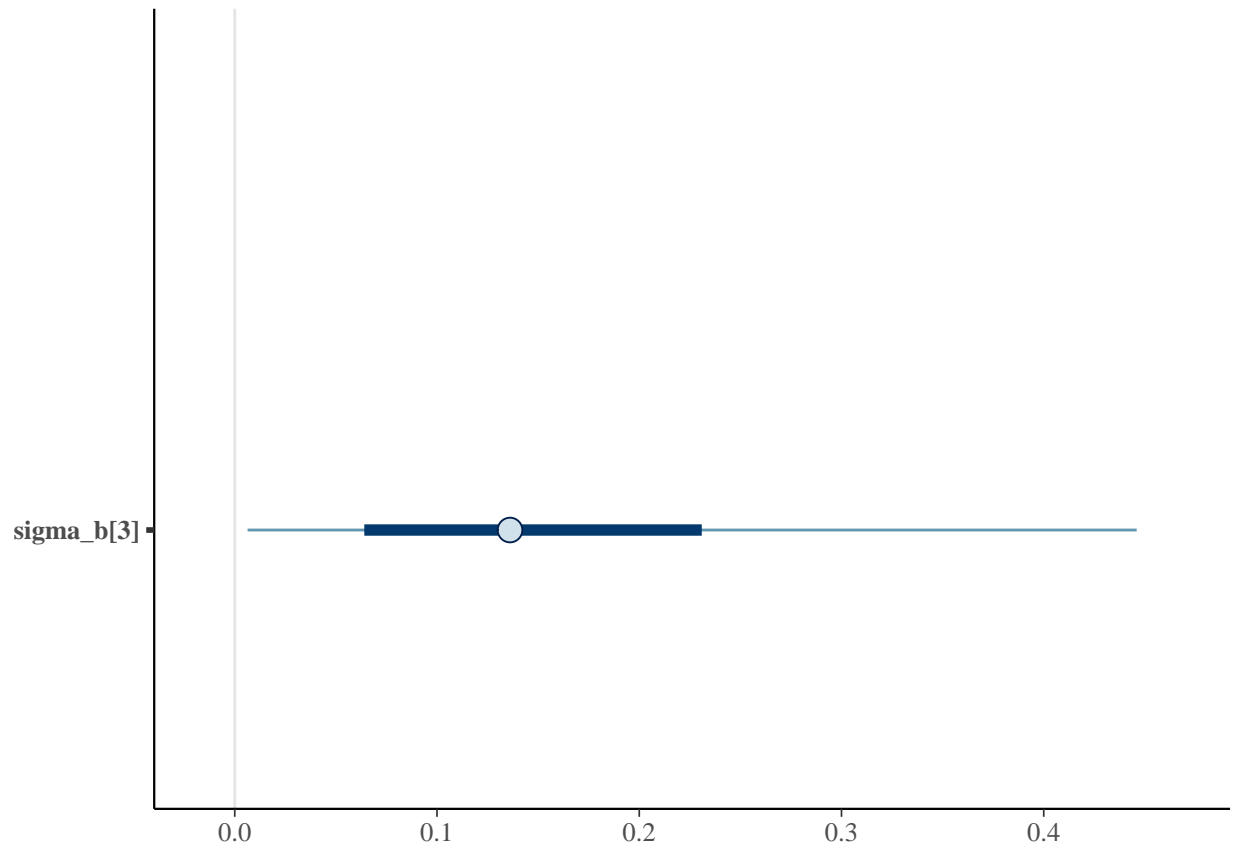
fsins_blmm <- mebn.get_localfit("fsins")

## [1] "Loading models\\fsins_blmm.rds"
# Index of predictor 'energy'
id <- match("energia", datadesc$Name)

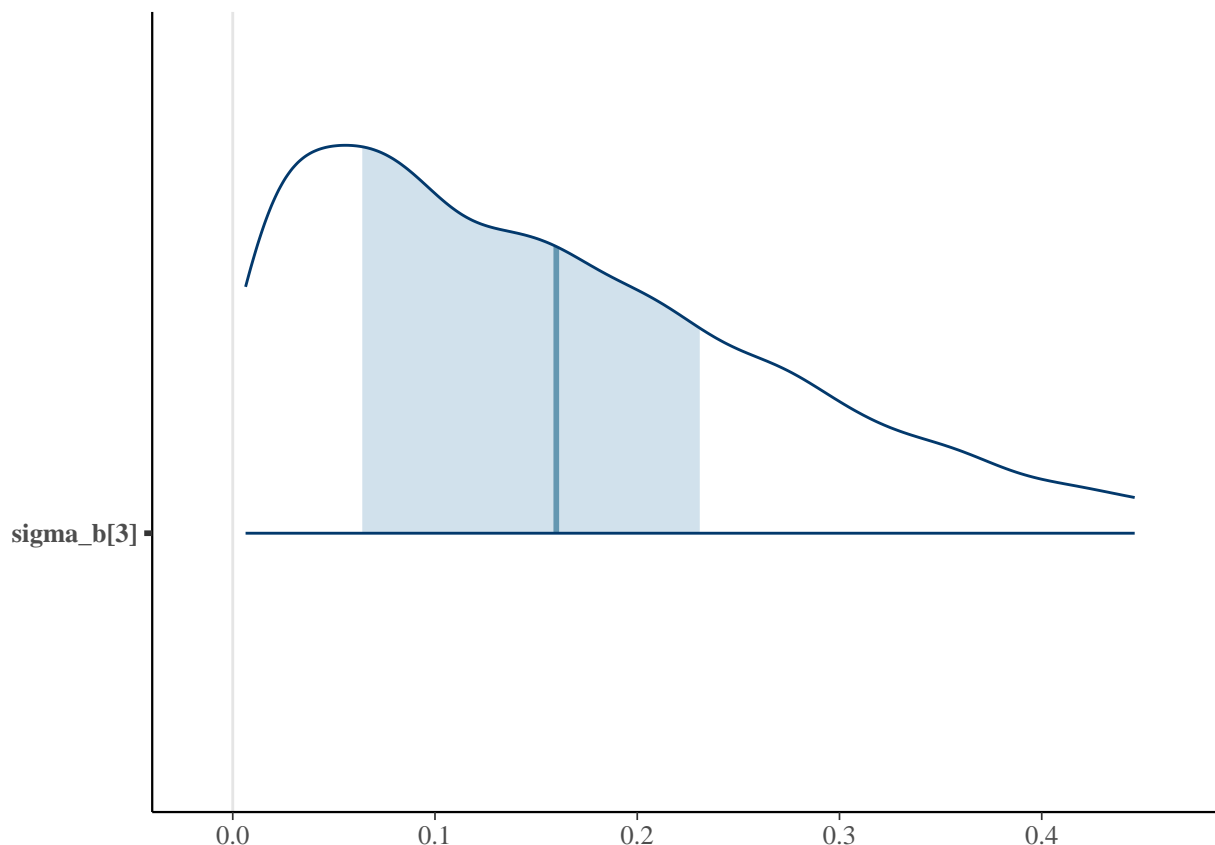
posterior <- as.array(fsins_blmm)

mcmc_intervals(posterior, pars = c(paste0("sigma_b[",id,"]")), prob_outer = 0.95)

```







This shows that there exists personal variance in how energy intake affects blood insulin levels as the 95% credible interval is above zero. The wide probability distribution shows that the exact estimate is quite uncertain and we would need more observations or stricter prior information for more precise estimation. One can also observe the difference between typical choices of Bayesian point estimation; the upper interval chart points at the posterior median, and the lower chart shows the posterior mean and MAP estimate at the high point of the probability.

## Conclusions

Mixed-effect Bayesian networks offer an appealing way to model the system of nutritional effects. By incorporating the mixed-effect modeling to local probability distributions, we can estimate the effects in both population and personal level. Furthermore, the fully Bayesian estimation of the distributions allows direct means for including prior information at the model and also addressing the uncertainty of the estimates.

After estimating the personal variances in population level, it is possible next to predict this kind of personal reaction graph for a new person with limited data only. These reaction graphs allow inference on any of the variables at the graph based on their predicted relationships. This enables interesting new applications in personal diet recommendations and personal health care.

## References

- Aussem, Alex, André Tchernof, Sérgio Rodrigues de Moraes, and Sophie Rome. 2010. "Analysis of Lifestyle and Metabolic Predictors of Visceral Obesity with Bayesian Networks." *BMC Bioinformatics* 11 (1): 487. doi:10.1186/1471-2105-11-487.
- Bae, Harold, Stefano Monti, Monty Montano, Martin H. Steinberg, Thomas T. Perls, and Paola Sebastiani.

2016. “Learning Bayesian Networks from Correlated Data” 6 (May). The Author(s) SN -: 25156 EP. <http://dx.doi.org/10.1038/srep25156>.
- Dawid, A. P., and S. L. Lauritzen. 1993. “Hyper Markov Laws in the Statistical Analysis of Decomposable Graphical Models.” *Ann. Statist.* 21 (3). The Institute of Mathematical Statistics: 1272–1317. doi:10.1214/aos/1176349260.
- Koller, Daphne, and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press.
- Lankinen, M., U. Schwab, M. Kolehmainen, J. Paananen, K. Poutanen, H. Mykkanen, T. Seppanen-Laakso, H. Gylling, M. Uusitupa, and M. Ore?i? 2011. “Whole grain products, fish and bilberries alter glucose and lipid metabolism in a randomized, controlled trial: the Sysdimet study.” *PLoS ONE* 6 (8): e22646.
- Nagarajan, Radhakrishnan, and Marco Scutari. 2013. *Bayesian Networks in R with Applications in Systems Biology*. New York: Springer. doi:10.1007/978-1-4614-6446-4.
- Piironen, Juho, and Aki Vehtari. 2017a. “Sparsity Information and Regularization in the Horseshoe and Other Shrinkage Priors.” *Electron. J. Statist.* 11 (2). The Institute of Mathematical Statistics; the Bernoulli Society: 5018–51. doi:10.1214/17-EJS1337SI.
- . 2017b. “Comparison of Bayesian Predictive Methods for Model Selection.” *Statistics and Computing* 27 (3): 711–35. doi:10.1007/s11222-016-9649-y.
- Scutari, Marco, and Jean-Baptiste Denis. 2014. *Bayesian Networks with Examples in R*. Boca Raton: Chapman; Hall.
- Zeevi, David, Tal Korem, Niv Zmora, David Israeli, Daphna Rothschild, Adina Weinberger, Orly Ben-Yacov, et al. 2015. “Personalized Nutrition by Prediction of Glycemic Responses.” *Cell* 163 (5). Elsevier: 1079–94. doi:10.1016/j.cell.2015.11.001.