Journal of Statistical Software

January 2018, Volume 83, Issue 1. *doi: 10.18637/jss.v083.i01*

meta4diag: Bayesian Bivariate Meta-Analysis of Diagnostic Test Studies for Routine Practice

Jingyi Guo
Norwegian University of
Science and Technology

Andrea Riebler
Norwegian University of
Science and Technology

Abstract

This paper introduces the R package **meta4diag** for implementing Bayesian bivariate meta-analyses of diagnostic test studies. Our package **meta4diag** is a purpose-built front end of the R package **INLA**. While **INLA** offers full Bayesian inference for the large set of latent Gaussian models using integrated nested Laplace approximations, **meta4diag** extracts the features needed for bivariate meta-analysis and presents them in an intuitive way. It allows the user a straightforward model specification and offers user-specific prior distributions. Further, the newly proposed penalized complexity prior framework is supported, which builds on prior intuitions about the behaviors of the variance and correlation parameters. Accurate posterior marginal distributions for sensitivity and specificity as well as all hyperparameters, and covariates are directly obtained without Markov chain Monte Carlo sampling. Further, univariate estimates of interest, such as odds ratios, as well as the summary receiver operating characteristic (SROC) curve and other common graphics are directly available for interpretation. An interactive graphical user interface provides the user with the full functionality of the package without requiring any R programming. The package is available from the Comprehensive R Archive Network (CRAN) at https://CRAN.R-project.org/package=meta4diag/ and its usage will be illustrated using three real data examples.

Keywords: Bayesian analysis, bivariate meta-analysis, diagnostic test studies, graphical user interface, integrated nested Laplace approximations, R package.

1. Introduction

A meta-analysis summarizes the results from multiple studies with the purpose of finding a general trend across the studies. It plays a central role in several scientific areas, such as medicine, pharmacology, epidemiology, education, psychology, criminology and ecology

(Borenstein, Hedges, Higgins, and Rothstein 2011). A bivariate meta-analysis of diagnostic test studies is a special type of meta-analysis that summarizes the results from separately performed diagnostic test studies while keeping the two-dimensionality of the data (Van Houwelingen, Arends, and Stijnen 2002; Reitsma, Glas, Rutjes, Scholten, Bossuyt, and Zwinderman 2005). Results of a diagnostic test study are commonly provided as a two-by-two table, which is a cross tabulation including four numbers: the number of patients tested positive that are indeed diseased (according to some gold standard), those tested positive that are not diseased, those tested negative that are however diseased and finally those tested negative that are indeed not diseased. Usually the table entries are referred to as true positives (TP), false positives (FP), false negatives (FN) and true negatives (TN), respectively. Those entries are used to compute pairs of sensitivity and specificity indicating the quality of the respective diagnostic test. The main goal of a bivariate meta-analysis is to derive summary estimates of sensitivity and specificity from several separately performed test studies. For this purpose pairs of sensitivity and specificity are jointly analyzed and the inherent correlation between them is incorporated using a random effects approach (Reitsma *et al.* 2005; Chu and Cole 2006). Related accuracy measures, such as likelihood ratios (LRs), which indicate the discriminatory performance of positive and negative tests, LR+ and LR− respectively, can be also derived. Further, frequently used estimates include diagnostics odds ratios (DORs) illustrating the effectiveness of the test or risk differences which are related to the Youden index (Altman 1990; Youden 1950).

Reitsma *et al.* (2005) proposed to model logit sensitivity and logit specificity using a bivariate normal likelihood, whereby the mean vector itself is modeled using a bivariate normal distribution (normal-normal model). Our new package **meta4diag** (Guo and Riebler 2017) follows the approach proposed by Chu and Cole (2006) and Hamza, Reitsma, and Stijnen (2008) using an exact binomial likelihood (binomial-normal model). This approach has been shown to outperform the approximate normal likelihood in terms of bias, mean-squared error (MSE) and coverage. Furthermore, it does not require a continuity correction for zero cells in the two-by-two table (Harbord, Deeks, Egger, Whiting, and Sterne 2007). Recently, Chen, Chu, Luo, Nie, and Chen (2015) and Kuss, Hoyer, and Solms (2014) proposed a third alternative, the beta-binomial model, where sensitivity and specificity are not modeled after the logit transformation but on the original scale using a beta distribution. The inherent correlation is then incorporated via copulas (Kuss *et al.* 2014). In the absence of covariates or in the case that all covariates affect both sensitivity and specificity (Harbord *et al.* 2007), the binomial-normal model can be reparameterized into the hierarchical summary receiver operating characteristic (HSROC) model (Rutter and Gatsonis 2001; Harbord *et al.* 2007). In contrast to the binomial-normal model the HSROC model uses a scale parameter and an accuracy parameter, which are functions of sensitivity and specificity and defines an underlying hierarchical SROC (summary receiver operating characteristic) curve.

Different statistical software environments, such as the SAS software (SAS Institute Inc. 2013), Stata (StataCorp. 2015) and R (R Core Team 2017), have been used in the past ten years to conduct bivariate meta-analysis of diagnostic test studies. Within a frequentist setting the SAS routines `PROC MIXED` and `PROC NLMIXED` can be used to fit the normal-normal and binomial-normal model, see for example Van Houwelingen *et al.* (2002); Arends, Hamza, Houwelingen, Heijenbrok-Kal, Hunink, and Stijnen (2008); Hamza, Arends, Van Houwelingen, and Stijnen (2009). The SAS macro **METADAS** provides a user-friendly interface for the binomial-normal model and the HSROC model (Takwoingi and Deeks 2008). Within Stata

the module **metandi** fits the normal-normal model using an adaptive quadrature (Harbord and Whiting 2009), while the module **mvmeta** performs maximum likelihood estimation of multivariate random-effects models using a Newton-Raphson procedure (White 2009; Gasparrini, Armstrong, and Kenward 2012). The R package **mada** (Doebler 2017), a specialized version of **mvmeta**, is specifically designed for the analysis of diagnostic accuracy. The package provides both univariate modeling of log odds ratios and bivariate binomial-normal modeling of sensitivity and specificity. A continuity correction is used for zero cells in the two-by-two tables.

Since the number of studies involved in a meta-analysis of diagnostic tests commonly is small, often less than 20 studies, and data within each two-by-two table can be sparse, the use of numerical algorithms for maximizing the likelihood of the above complex bivariate model might be problematic and lead to non-convergence (Paul, Riebler, Bachmann, Rue, and Held 2010). Bayesian inference that introduces prior information for the variance and correlation parameters in the bivariate term is therefore attractive (Harbord 2011). Markov chain Monte Carlo (MCMC) algorithms can be implemented through the generic frameworks **WinBUGS** (Lunn, Thomas, Best, and Spiegelhalter 2000), **OpenBUGS** (Lunn, Spiegelhalter, Thomas, and Best 2009) or JAGS (Plummer 2003). There exist further specialized R packages for analyzing diagnostic test studies in Bayesian setting, such as **bamdit** or **HSROC** (Verde 2017; Schiller and Dendukuri 2015). Instead of modeling the link-transformed sensitivity and specificity directly, the package **bamdit** models the differences (D_i) and sums (S_i) of the link-transformed sensitivity and specificity jointly. The quantities D_i and S_i are roughly independent by using these linear transformations, so that Verde (2010) used a zero centered prior for the correlation of D_i and S_i to represent vague prior information. Consequently, JAGS is used for model estimation. In contrast, package **HSROC** builds on the HSROC model to jointly analyze sensitivity and specificity with and without a gold standard reference test. Uniform priors on a restricted interval are thereby assumed for all the hyperparameters and model estimation is carried out using a Gibbs sampler (Chen and Peace 2013, Chapter 10). However, the use of Bayesian approaches is still limited in practice which might be partly caused by the fact that many applied scientists feel not comfortable with using MCMC sampling-based procedures (Harbord 2011). Implementation needs to be performed carefully to ensure mixing and convergence. Furthermore, MCMC based methods are often time consuming, in particular, when interest lies in simulation studies which require several MCMC runs.

Paul *et al.* (2010) proposed to perform full Bayesian inference using integrated nested Laplace approximations (INLA) which avoids MCMC entirely (Rue, Martino, and Chopin 2009). The R package **INLA** (Lindgren and Rue 2015), see `http://www.R-INLA.org/`, implements Bayesian inference using INLA for the large set of latent Gaussian models. However, we understand that the range of options and the required knowledge of available features in **INLA** might be overwhelming for the applied user interested in only one specific model. Here, we present a new R package **meta4diag** which is a purpose-built package defined on top of **INLA** extracting only the features needed for bivariate meta regression. Our package **meta4diag** implements the binomial-normal model. Model definition is straightforward, and output statistics and graphics of interest are directly available. Therefore, users do not need to know the structure of the general **INLA** output object. Although its greatest strength, another criticism towards Bayesian inference is the choice of prior distributions. Our package **meta4diag** allows the user to specify prior distributions for the hyperparameters using intuitive statements based on the recently proposed framework of penalized complexity (PC) priors (Simpson, Rue, Riebler,

Martins, and Sørbye 2017). Alternatively, standard prior distributions or user-specific prior distributions can be used. Our package is appealing for routine use and applicable without any deep knowledge of the programming language R via the integrated graphical user interface (GUI) offering roll-down menus and dialog boxes implemented using the R package **shiny** (Chang, Cheng, Allaire, Xie, and McPherson 2017).

The rest of this paper is organized as follows. In Section 2 we introduce the binomial-normal model and discuss its estimation within a Bayesian inference setting. Here, specific emphasis is given on the definition of prior distributions. Section 3 illustrates the functionality of the package **meta4diag**. Model output and available graphics are described based on the previously analyzed `Telomerase` (Glas, Roos, Deutekom, Zwinderman, Bossuyt, and Kurth 2003), `Scheidler` (Scheidler, Hricak, Yu, Subak, and Segal 1997) and `Catheter` (Chu, Guo, and Zhou 2010) data sets. Further, the user-friendly graphical user interface is presented. Finally, a conclusion is given in Section 4.

2. Introducing the statistical framework

2.1. Binomial-normal model for bivariate meta-analysis

In a bivariate meta-analysis, each study presents the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). Let $\text{Se} = \text{TP}/(\text{TP} + \text{FN})$ denote the true positive rate (TPR) which is known as sensitivity and $\text{Sp} = \text{TN}/(\text{TN} + \text{FP})$ the true negative rate (TNR) which is known as specificity. Chu and Cole (2006) proposed the following bivariate generalized linear mixed effects model to summarize the results of several diagnostic studies, $i = 1, \ldots, I$, by modeling sensitivity and specificity jointly:

$$\begin{aligned} \text{TP}_i|\text{Se}_i &\sim \text{Binomial}(\text{TP}_i + \text{FN}_i, \text{Se}_i), & \text{logit}(\text{Se}_i) &= \mu + \mathbf{U}_i\boldsymbol{\alpha} + \phi_i, \\ \text{TN}_i|\text{Sp}_i &\sim \text{Binomial}(\text{TN}_i + \text{FP}_i, \text{Sp}_i), & \text{logit}(\text{Sp}_i) &= \nu + \mathbf{V}_i\boldsymbol{\beta} + \psi_i, \\ \begin{pmatrix}\phi_i \\ \psi_i\end{pmatrix} &\sim \mathcal{N}\left[\begin{pmatrix}0 \\ 0\end{pmatrix}, \begin{pmatrix}\sigma_\phi^2 & \rho\sigma_\phi\sigma_\psi \\ \rho\sigma_\phi\sigma_\psi & \sigma_\psi^2\end{pmatrix}\right]. \end{aligned} \tag{1}$$

Here, μ, ν denote the intercepts for $\text{logit}(\text{Se}_i)$ and $\text{logit}(\text{Sp}_i)$, respectively, and $\mathbf{U}_i$, $\mathbf{V}_i$ study-level covariate vectors with corresponding coefficient parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The covariance matrix of the random effects ϕ_i and ψ_i is parameterized using between-study variances σ_ϕ^2, σ_ψ^2 and correlation ρ.

The most-commonly-used logit link function can be replaced by other monotone link functions, such as the probit or the complementary log-log transformation. We assume that both sensitivity and specificity are modeled with the same link function. If desired, model (1) can easily be changed to model sensitivity and the false positive rate $(1-\text{Sp})$, or the false negative rate $(1-\text{Se})$ and specificity, or $1-\text{Se}$ and $1-\text{Sp}$, instead of sensitivity and specificity, causing the corresponding change in parameter estimates. Different model options are available through the argument `model.type` in the package **meta4diag**, see Section 3.3.

2.2. Specification of prior distributions

We specify prior distributions for all parameters, i.e., the three hyperparameters σ_ϕ^2, σ_ψ^2 and ρ, as well as the fixed effects μ, ν, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Per default a normal prior with zero mean and

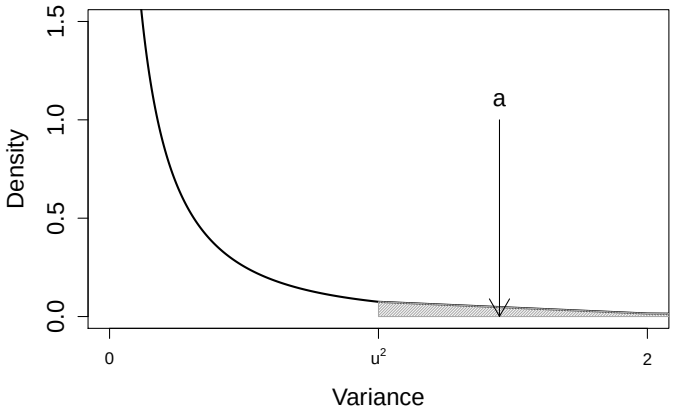


Figure 1: An example of the PC prior for the variance calibrated such that $\mathsf{P}(\sigma > 1) = 0.05$. The black line is the prior density and the shaded area denotes the density weight $a = 0.05$ when the standard deviation is larger than $u = 1$.

large variance is used for the fixed effects μ, ν, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The user is free to specify any prior distribution for σ^2_ϕ, σ^2_ψ and ρ including the newly proposed penalized complexity (PC) priors, see Simpson *et al.* (2017) for details. One of the four principles underlying PC priors is Occam's razor. The idea is to see a certain model component as a flexible extension of a base model (commonly a simpler model) to which we would like to reduce if not otherwise indicated by the data. Thinking of a Gaussian random effect with mean zero and covariance matrix $\sigma^2\mathbf{I}$, the base model would be $\sigma^2 = 0$, i.e., the absence of the effect. A PC prior puts maximum density mass at the base model and decreasing mass with increasing distance away from the base model. The PC prior for the variance components σ^2_ϕ or σ^2_ψ is discussed in Simpson *et al.* (2017, Section 2.3) and corresponds to an exponential prior with parameter λ for the standard deviation σ_ϕ or σ_ψ, respectively. A simple choice to set λ is to provide (u, a) such that $\mathsf{P}(\sigma > u) = a$ leading to $\lambda = -\log(u)/a$ with $u > 0$ and $0 < a < 1$. Figure 1 shows an example of the PC prior for the variance. In practice, the PC prior for the variance parameter in a diagnostic meta-analysis could be derived from the belief of the interval that sensitivities or specificities lie in. For example, choosing the contrast $\mathrm{P}(\sigma > 3) = 0.05$ corresponds to believing that the sensitivities or specificities lie in the interval $[0.5, 0.95]$ with probability 0.95 (Wakefield 2007).

For the correlation parameter ρ, Harbord (2011) proposed to use a stronger prior than the normal prior for the Fisher's z-transformed correlation, which was used in Paul *et al.* (2010). Motivated by the nature of diagnostic tests he proposed to use a prior which is not centered around zero but defined around some (negative) base value ρ_0 instead (Reitsma *et al.* 2005). Using the PC prior framework the above suggestions can be implemented directly. Simpson *et al.* (2017, Appendix A.3) derives the PC prior for the correlation parameter in an autoregressive model of first order assuming the base model being defined at $\rho_0 = 0$ and identical statistical behavior left and right of 0. Although slightly tedious, this derivation can be generalized to an arbitrary ρ_0 and asymmetrical behavior to the left and right of ρ_0 (Guo, Riebler, and Rue 2017). Within `meta4diag()` we offer three strategies to intuitively define a PC prior for ρ given an arbitrary value of ρ_0. Similar as for the variance, probability contrasts are used to define the prior intuitively.

Strategy 1: Specify the left tail behavior and the probability mass on the left-hand side of ρ_0 by,

$$\mathsf{P}(\rho < u_1|\rho_0) = a_1 \qquad \text{and} \qquad \mathsf{P}(\rho < \rho_0) = \omega.$$

Here, $(\rho_0, \omega, u_1, a_1)$ are the hyperparameters needed to define the prior density.

Strategy 2: Specify the right tail behavior and the probability mass on the left-hand side of ρ_0 by,

$$\mathsf{P}(\rho > u_2|\rho_0) = a_2 \qquad \text{and} \qquad \mathsf{P}(\rho < \rho_0) = \omega.$$

Here, $(\rho_0, \omega, u_2, a_2)$ are the hyperparameters needed to define the prior density.

Strategy 3: Specify left and right tail behaviors, by

$$\mathsf{P}(\rho < u_1|\rho_0) = a_1 \qquad \text{and} \qquad \mathsf{P}(\rho > u_2|\rho_0) = a_2.$$

Here, $(\rho_0, u_1, a_1, u_2, a_2)$ are the hyperparameters needed to define the prior density.

Figure 2 shows examples of the PC prior for the correlation using the three different strategies. The prior density used in Paul *et al.* (2010) is shown as the gray dashed lines for comparison. The parameters for the strategies are motivated based on the estimation results from Menke (2014), who analyzed 50 independent bivariate meta-analyses which were selected randomly from the literature within a Bayesian setting, and Diaz (2015), who reported frequentist estimates based on a literature review of 61 bivariate meta-analyses of diagnostic accuracy published in 2010. According to these two publications, the distribution of the correlation seems asymmetric around zero. We find that around half of the correlation point estimates are negative, with a mode around -0.2. Only a small proportion are larger than 0.4 and values larger than 0.8 are rare. Based on these findings, we choose three differently behaved PC priors that are all defined around $\rho_0 = -0.2$.

Defining the parameters of the prior distributions based on probability contrasts seems very intuitive. As illustrated it is straightforward to incorporate available prior knowledge into the prior distributions, while still having the option to define vague priors using less stringent probability contrasts. Although we recommend to specify priors for the variance and correlation components separately, our package also offers the option to use an inverse Wishart distribution as a prior for the entire covariance matrix.

3. Using package meta4diag

3.1. Package overview

The **meta4diag** package provides functions for fitting bivariate meta-analyses within a full Bayesian setting as outlined in Section 2. The package is available via the Comprehensive R Archive Network (CRAN) at https://CRAN.R-project.org/package=meta4diag and can be directly installed in R by typing

```
R> install.packages("meta4diag")
```

(given a working internet connection and the appropriate access rights on the computer). Within this paper we use package version 2.0.7 and **INLA** version 17.11.11. Of note is that **meta4diag** requires **INLA** to be installed, which can be done using

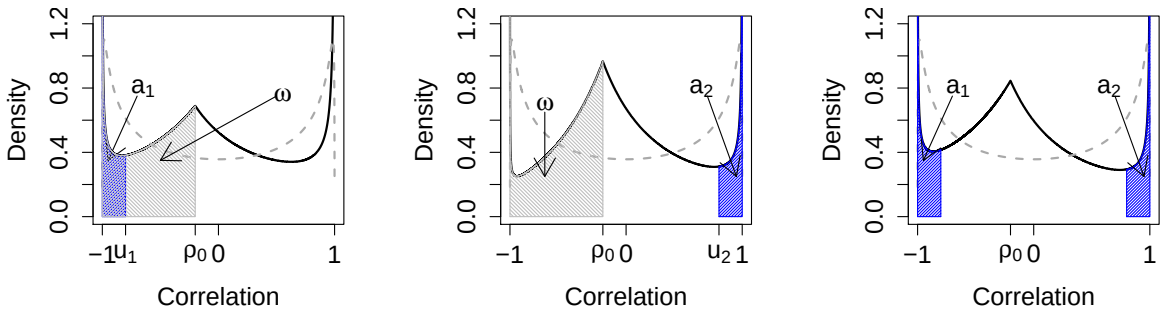


Figure 2: Illustration of potential PC priors for the correlation parameter ρ. The black solid line shows the PC prior and the dashed gray line shows the prior density proposed by Paul *et al.* (2010). In all plots we use $\rho_0 = -0.2$. (a) Strategy 1: Prior derived using $\mathsf{P}(\rho < -0.8|\rho_0 = -0.2) = 0.1$ and $\mathsf{P}(\rho < -0.2) = 0.4$. (b) Strategy 2: Prior derived using $\mathsf{P}(\rho > 0.8|\rho_0 = -0.2) = 0.1$ and $\mathsf{P}(\rho < -0.2) = 0.4$. (c) Strategy 3: Prior derived using $\mathsf{P}(\rho < -0.8|\rho_0 = -0.2) = 0.1$ and $\mathsf{P}(\rho > 0.8|\rho_0 = -0.2) = 0.1$.

```
R> install.packages("INLA",
+    repos = "https://inla.R-INLA-download.org/R/testing",
+    dependencies = TRUE)
```

Once the package and its dependencies are installed all analyses presented throughout this work are reproducible.

The **meta4diag** package consists of one major function called `meta4diag()`. This function estimates the Bayesian bivariate regression model for diagnostic test studies, assuming each study provides TP, FP, TN and FN. Several studies can be grouped according to a categorical variable. Posterior estimates for parameters of the bivariate model as well as common plots and summary statistics are directly available. Inference is thereby performed using **INLA**, which provides accurate deterministic approximations to all model parameters and linear summary estimates. Based on the output of `meta4diag()` different plots of interest can be generated and also non-linear summary estimates, for example the diagnostics odds ratio (DOR), are available based on Monte Carlo estimation, whereby i.i.d. samples are generated from the approximated posterior distribution using a built-in function of **INLA**.

The package includes three data sets which will be used in the following subsections to illustrate the functionality of **meta4diag**. The data sets differ in their structure and the availability of covariates. The first data set, called `Telomerase`, was presented by Glas *et al.* (2003) and consists of 10 diagnostic test studies. There is no covariate information available. The low number of studies involved makes this data set challenging when using maximum likelihood procedures, see for example Riley, Abrams, Sutton, Lambert, and Thompson (2007) and Paul *et al.* (2010). The second data set, called `Scheidler`, was presented in Scheidler *et al.* (1997) and combines three meta-analyses to compare the utility of three types of diagnostic imaging procedures to detect lymph node metastases in patients with cervical cancer. The third data set, called `Catheter`, consists of 33 studies from a diagnostic accuracy analysis presented by Chu *et al.* (2010) and provides disease prevalence as additional covariate.

3.2. General data structure required

The first argument `data` in the function `meta4diag()` is the data set. It should be given as a data frame with a minimum of 4 columns named `TP`, `FP`, `TN` and `FN`. If there is no column named `studynames` providing study names, the `meta4diag()` function will generate an additional column setting the study name indicators to `study_1`, ..., `study_n`, where n is the number of studies in the meta-analysis. Further columns are considered to be covariates. The data set `Telomerase` can thus be defined using five columns, where the first column provides study name indicators and the remaining four provide values of TP, FP, TN and FN.

```
R> studynames <- c("Ito_1998", "Rahat_1998", "Kavaler_1998", "Yoshida_1997",
+    "Ramakumar_1999", "Landman_1998", "Kinoshita_1997", "Gelmini_2000",
+    "Cheng_2000", "Cassel_2001")
R> TP <- c(25, 17, 88, 16, 40, 38, 23, 27, 14, 37)
R> FP <- c(1, 3, 16, 3, 1, 6, 0, 2, 3, 22)
R> TN <- c(25, 11, 31, 80, 137, 24, 12, 18, 29, 7)
R> FN <- c(8, 4, 16, 10, 17, 9, 19, 6, 3, 7)
R> Telomerase <- data.frame(studynames = studynames, TP = TP, FP = FP,
+    TN = TN, FN = FN)
R> head(Telomerase)

      studynames TP FP  TN FN
1       Ito_1998 25  1  25  8
2     Rahat_1998 17  3  11  4
3   Kavaler_1998 88 16  31 16
4   Yoshida_1997 16  3  80 10
5 Ramakumar_1999 40  1 137 17
6   Landman_1998 38  6  24  9
```

3.3. Analyzing a standard meta-analysis without covariate information

Here, we show how to analyze the `Telomerase` data set which represents a meta-analysis of studies that use the telomerase marker for the analysis of bladder cancer. To analyze the data set, we first load the **INLA** and the **meta4diag** package in R using:

```
R> library("INLA")
R> library("meta4diag")
```

We then call the function `meta4diag()` as follows:

```
R> set.seed(18674)
R> res <- meta4diag(data = Telomerase, model.type = 1, var.prior = "PC",
+    var2.prior = "PC", cor.prior = "Normal", var.par = c(3, 0.05),
+    cor.par = c(0, 5), link = "logit", nsample = 10000, seed = 1672)
```

The data set is transferred as the first argument followed by the argument `model.type = 1`, saying that we would like to model sensitivity and specificity jointly. Of note is that

the argument `model.type` can be any integer from 1 to 4 depending on which two accuracy measures are going to be modeled. When `model.type = 1`, sensitivity and specificity are modeled jointly. The sensitivity and (1 – specificity), (1 – sensitivity) and specificity and (1 – sensitivity) and (1 – specificity) will be jointly modeled when `model.type = 2`, `model.type = 3` and `model.type = 4`, respectively. The argument `var.prior` is a character string to specify the prior distribution for the (transformed) variance component of the first accuracy measure, i.e., here the sensitivity. The options are `"PC"` for the PC prior, `"Tnormal"` for the truncated normal prior, `"Hcauchy"` for the half-Cauchy prior and `"Unif"` for the uniform prior, which are all defined on the standard deviation scale. Alternatively `"Invgamma"` for the inverse gamma prior or any user specified prior defined on the variance scale can be chosen. A user-specified prior for the variance is chosen by setting `var.prior = "Table"` and providing a 2-column data frame to `var.par`. The first column provides support points for the variance which should be in $[0, \infty]$, and the second column provides the corresponding prior density at these points. Of note it that the usage of the `"Table"` prior in **meta4diag** is different from that in **INLA**. While **INLA** requires the user to define the `"Table"` prior on the internal parameterization of the hyperparameter, the user of **meta4diag** can work on the original scale. The argument `var2.prior` is a character string to specify the prior distribution for the second variance component. The options are the same as for the argument `var.prior`.

The argument `cor.prior` is a character string defining the prior distribution for the (transformed) correlation parameter between the two accuracy measures. The options are `"PC"` for the PC prior defined on the correlation scale, `"Normal"` for the normal distribution defined on the Fisher's z-transformed correlation, `"Beta"` for the beta distribution defined on a suitable transformation, see documentation, and `"Table"` for an user specific prior defined on the correlation scale. The `"Table"` prior for the correlation should be provided as a 2-column data frame, where the first column provides suitable support points within $[-1, 1]$, and the second column provides the corresponding density mass of those points. Alternatively, if at least one of the three arguments `var.prior`, `var2.prior` and `cor.prior` is set to `"Invwishart"`, an inverse Wishart distribution will be used for the covariance matrix ignoring any other prior definitions for the remaining arguments. The arguments `var.par`, `var2.par`, `cor.par` are numerical vectors specifying the hyperparameters for the priors for variance and correlation parameters. If the inverse Wishart prior is used the hyperparameters can be set in `wishart.par`. Prior definitions including parameterizations of the different options are given in the package documentation of `meta4diag()` or `makePriors()`. Of note is that the arguments `var.prior`, `var2.prior` and `cor.prior` are not case sensitive, i.e., `var.prior = "pc"` is valid if one uses it to indicate the PC prior for the first variance component.

Here, we use the logit link function by using `link = "logit"`. Alternative options are `"probit"` for the probit link and `"cloglog"` for the complementary log-log transformation. The argument `quantiles` requires a numerical vector with values in $[0, 1]$ defining which posterior quantiles should be returned. The default setting is `c(0.025, 0.5, 0.975)`, and these three quantiles will always be returned. The argument `nsample` is an integer specifying the number of i.i.d. samples, generated from the approximated posterior distribution, which are used to compute any non-linear function of interest, such as DOR, LR+ or LR−. The argument `seed` is required when `nsample` > 0 and used to control the random number generator for sampling from the posterior distributions in **INLA**. In order to reproduce the result, we also need to control the seed for the random number generator in R by controlling the variable `.Random.seed` or using the function `set.seed`.

To get summary information for all parameters of the model, we use the function `summary()`:

```
R> summary(res)

Time used:
 Pre-processing    Running inla Post-processing           Total
      1.9662211       0.2238450       0.3630319       2.5530980

Fixed effects:
    mean    sd 0.025quant 0.5quant 0.975quant
mu 1.192 0.198      0.806    1.190      1.595
nu 2.302 0.651      1.073    2.      3.679

Model hyperpar:
           mean    sd 0.025quant 0.5quant 0.975quant
var_phi   0.243 0.178      0.050    0.195      0.717
var_psi   3.648 2.073      1.148    3.142      9.082
cor      -0.819 0.200     -0.992   -0.888     -0.244

--------------------
          mean    sd 0.025quant 0.5quant 0.975quant
mean(Se) 0.766 0.032      0.699    0.767      0.825
mean(Sp) 0.897 0.052      0.767    0.907      0.971

--------------------
Correlation between mu and nu is -0.5504.
Marginal log-likelihood: -65.0459
Variable names for marginal plotting:
      mu, nu, var1, var2, rho
```

Here, also the time needed to fit the model as well as the estimated correlation between the two linear predictors, here μ and ν, are shown. This correlation is different from the hyper-parameter correlation provided in `cor`, which corresponds to $\hat{\rho}$, i.e., the posterior correlation between random effects.

To plot the posterior marginal distribution of σ^2_ϕ, say, we call the function `plot()` with argument `var.type = "var1"`. When defining separate prior distributions for the variance and correlation parameters and setting `overlay.prior = TRUE` the prior distribution is shown in the same device. The posterior marginal distributions of σ^2_ϕ and σ^2_ψ together with their corresponding prior distribution are shown in Figure 3. Valid values of `var.type` are the names of the fixed effects (i.e., `"mu"` and `"nu"` for this data set), `"var1"`, `"var2"` or `"rho"`. The argument `save` can be set to `FALSE` (default) to indicate that resulting figures are not saved on the computer, or to a file name, (e.g., `"posterior_v1.pdf"`), to indicate that the plot is saved as `"./meta4diagPlot/posterior_v1.pdf"`, where `"./"` denotes the current working directory and the directory `meta4diagPlot` is created automatically if it does not exist. Alternatively, the argument `save` can be set to `TRUE` to indicate that the plot is saved in the directory `meta4diagPlot` whereby the name is chosen according to `var.type`. Many

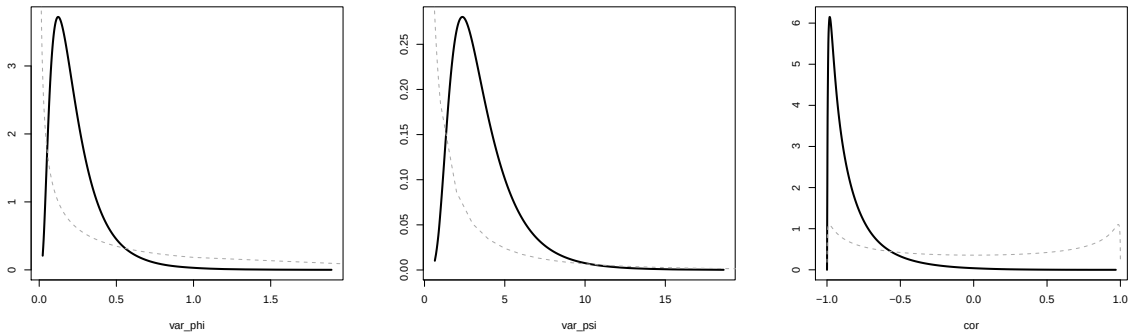


Figure 3: Posterior marginals (black solid line) of σ_ϕ^2, σ_ψ^2 and ρ for the `Telomerase` data together with the prior distributions (gray dashed line).

standard R plotting arguments, such as `xlab`, `ylab`, `xlim`, `ylim` and `col`, can also be set in the `plot()` function.

```
R> par(mfrow = c(1, 3))
R> plot(res, var.type = "var1", overlay.prior = TRUE, lwd = 2, save = FALSE)
R> plot(res, var.type = "var2", overlay.prior = TRUE, lwd = 2, save = FALSE)
R> plot(res, var.type = "rho", overlay.prior = TRUE, lwd = 2, save = FALSE)
```

To get descriptive statistics of study-specific accuracy measures of interest, such as positive or negative likelihood ratios LR+ or LR−, or the diagnostic odds ratio DOR, we call the function `fitted()`. The argument `accuracy.type` requires a single character string specifying the statistics of interest. Possible options are besides other `"sens"` (default), `"spec"`, `"TPR"`, `"TNR"`, `"FPR"`, `"FNR"`, `"LRpos"`, `"LRneg"`, `"RD"`, `"DOR"`, `"LLRpos"`, `"LLRneg"` and `"LDOR"` .

```
R> fitted(res, accuracy.type = "TPR")

Diagnostic accuracies true positive rate (sensitivity):
                 mean      sd 0.025quant 0.5quant 0.975quant
Ito_1998       0.7405 0.04928     0.6348   0.7433     0.8306
Rahat_1998     0.7938 0.04631     0.6929   0.7966     0.8782
Kavaler_1998   0.8294 0.02931     0.7699   0.8301     0.8851
Yoshida_1997   0.6936 0.05958     0.5562   0.7010     0.7910
Ramakumar_1999 0.6878 0.04830     0.5875   0.6898     0.7777
Landman_1998   0.7967 0.03804     0.7164   0.7982     0.8677
Kinoshita_1997 0.6219 0.06810     0.4784   0.6263     0.7418
Gelmini_2000   0.7797 0.04416     0.6879   0.7810     0.8639
Cheng_2000     0.7725 0.04940     0.6672   0.7744     0.8664
Cassel_2001    0.8550 0.03843     0.7703   0.8582     0.9212

R> fitted(res, accuracy.type = "DOR")

Diagnostic accuracies diagnostic odds ratio (DOR):
                  mean       sd 0.025quant 0.5quant 0.975quant
Ito_1998        70.720   66.770    13.4700   52.900    236.000
```

```
Rahat_1998       19.260   12.510     5.0830   16.210     51.460
Kavaler_1998     10.480    3.766     5.0000    9.843     19.420
Yoshida_1997     72.930   44.170    20.7900   62.500    186.600
Ramakumar_1999 211.700  190.300    47.3400  156.300    687.500
Landman_1998     18.750    8.933     6.9920   16.930     41.290
Kinoshita_1997 720.400 5751.000    12.4000  145.400   4412.000
Gelmini_2000     36.120   26.130     9.3120   29.460    102.700
Cheng_2000       37.470   22.420    11.0100   32.070     96.720
Cassel_2001       2.798    1.443     0.9042    2.494      6.474
```

A commonly used graphic to illustrate the results of a meta-analysis is the so-called forest plot (Lewis and Clarke 2001). Figure 4 shows the forest plot including 95% credible intervals for the `Telomerase` data set as obtained using the `forest()` function.

```
R> forest(res, accuracy.type = "sens", est.type = "mean", cut = c(0.4, 1),
+    nameShow = TRUE, dataShow = "center", estShow = TRUE, text.cex = 1.5,
+    arrow.lwd = 1.5)
```

The arguments `nameShow`, `dataShow`, `estShow` require a logical value indicating whether the study names, the given observations (values of TP, FP, TN and FN) and values of credible intervals are displayed as texts in the forest plot, respectively. The corresponding texts are right aligned when the arguments are set to be `TRUE`. They could also be `"left"`, `"right"` or `"center"` specifying the different alignments. The argument `accuracy.type` is defined as in the `fitted()` function. The argument `est.type` requires a character string specifying the summary estimate to be used. The options are `"mean"` (default) and `"median"`. The arguments `text.cex` specifies the text size, while `arrow.lwd` specifies the line width for the credible lines.

The two functions `crosshair()` and `SROC()` are available to study the result in the ROC space with sensitivity on the y-axis and 1 − specificity on the x-axis. Figure 5 shows a crosshair plot displaying the individual studies in ROC space with paired confidence intervals representing sensitivity and specificity (Phillips, Stewart, and Sutton 2010). Figure 6 shows a summary receiver operating characteristic curve (SROC) which is only available when no separate covariates are included for the two model components, here sensitivity and specificity, as only then the bivariate meta-regression approach is equivalent to the HSROC approach (Rutter and Gatsonis 2001). The corresponding commands are:

```
R> crosshair(res, est.type = "mean", col = 1:10)
R> SROC(res, est.type = "mean", sroc.type = 1, dataShow = "o",
+    crShow = TRUE, prShow = TRUE)
```

The argument `dataShow` specifies whether the original data are shown. The argument `crShow` and `prShow` are Boolean and indicate whether a credible region or prediction region, respectively, is shown. The argument `sroc.type` takes an integer value from 1 to 5. When `sroc.type = 1`, the function used to define the SROC line corresponds to "The regression line 1" in Arends *et al.* (2008); Chappell, Raab, and Wardlaw (2009). The values `sroc.type = 2`, `sroc.type = 3`, `sroc.type = 4` and `sroc.type = 5` correspond to "The major axis method", "The Moses and Littenberg's regression line", "The regression line 2" and "The

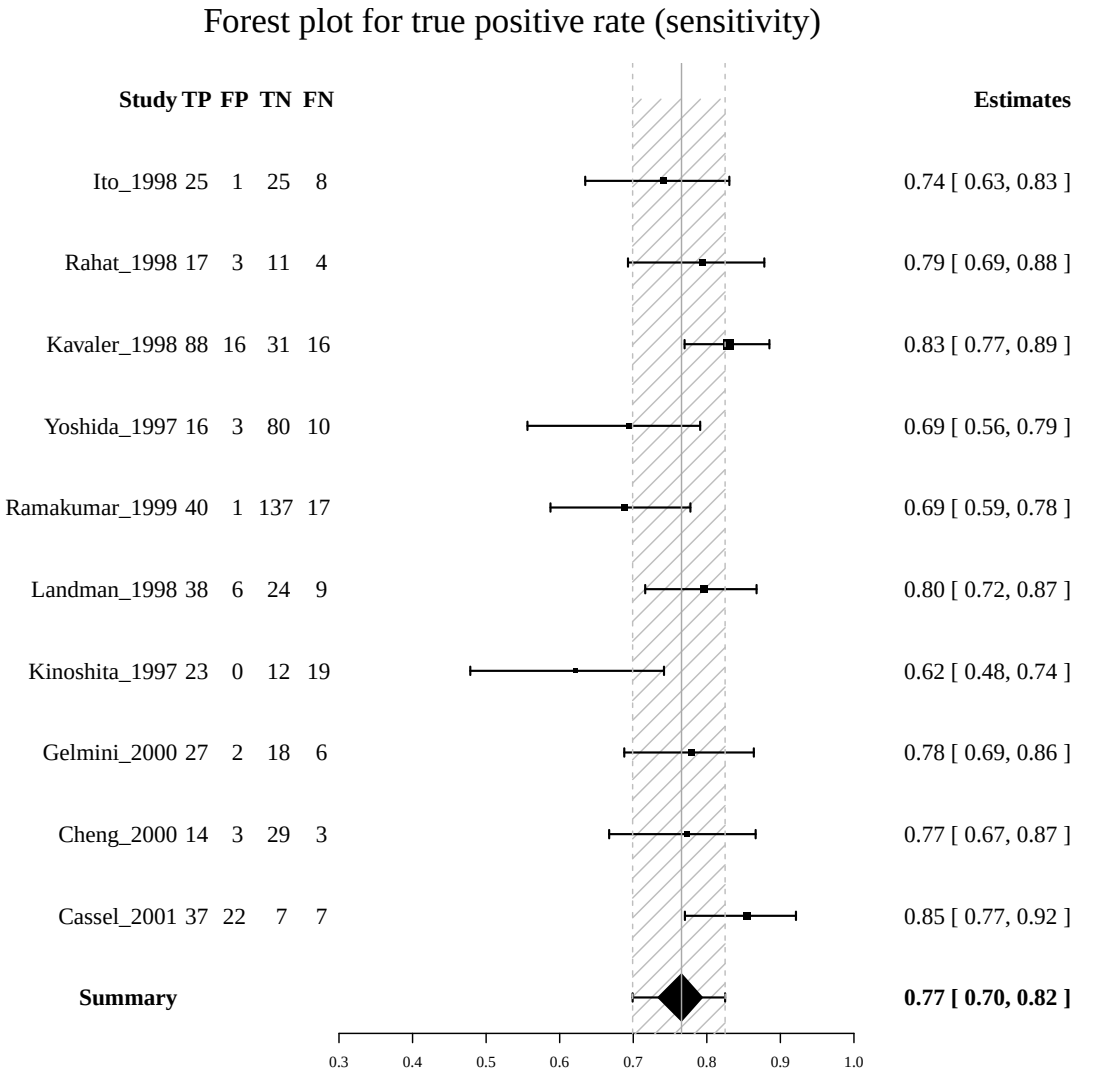


Figure 4: Forest plot of the true positive rate (sensitivity) for the `Telomerase` data. Study names, the given observations (values of TP, FP, TN and FN) as well as model-based mean estimates within 95% credible intervals are shown. At the bottom a summary estimate combining all studies is provided. The size of the study specified estimate points (■) is proportional to the length of the corresponding credible intervals, the shorter the interval length the bigger the point and vice versa.

Rutter and Gatsonis's SROC curve", respectively. Different choices may result in different SROC lines when the correlation for sensitivity and specificity is positive. We refer to Chappell *et al.* (2009) for more details and a comparison of the different formulations.

3.4. Incorporating additional sub-data stratification

The `Scheidler` data set contains the results of a meta-analysis conducted by Scheidler *et al.* (1997) to compare the utility of three types of diagnostic imaging, lymphangiography (LAG), computed tomography (CT) and magnetic resonance (MRI), to detect lymph node metastases

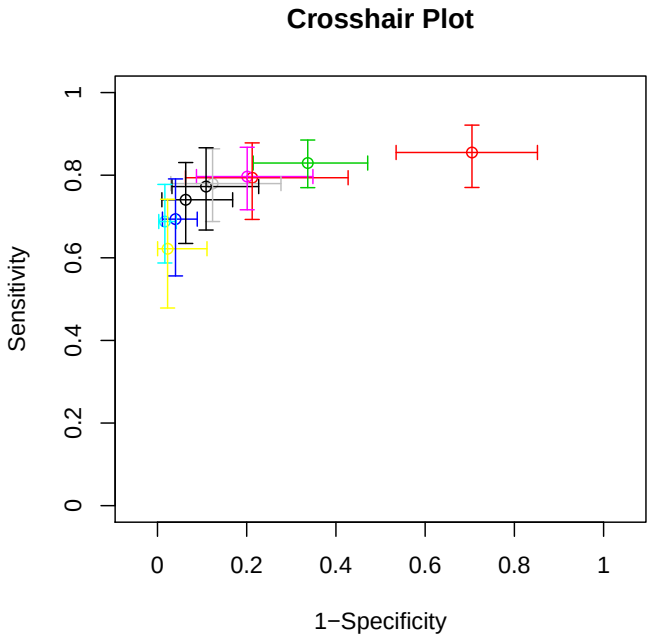


Figure 5: Crosshair plot for the `Telomerase` data set. Shown are the posterior means for each study as the summary points together with paired lines showing the corresponding 95% credible intervals for sensitivity and (1 – specificity). Colors are randomly chosen.

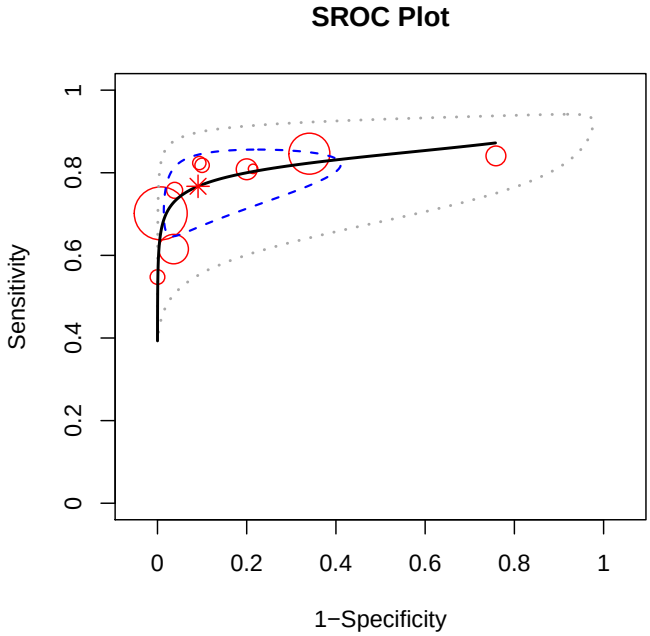


Figure 6: SROC plot for the `Telomerase` data set. Each bubble represents one study and indicates its observed sensitivity and specificity. The size of the bubble is proportional to the number of individuals in this study. The solid black line is the SROC line. The star point represents the summary point, the dashed blue line is the 95% credible region and the dashed gray line is the 95% prediction region.

in patients with cervical cancer. The data set consists of a total of 44 studies: the first 17 for CT, the following 17 for LAG and the last 10 for MRI. The `Scheidler` data set is provided in the package as a data frame with 44 rows. It has a special column named `modality` that specifies to which imaging technology, namely CT, LAG or MRI, each study belongs to. The first five lines of the data set are given as:

```
R> data("Scheidler", package = "meta4diag")
R> head(Scheidler)
```

```
           studynames modality TP FP FN TN
1       Grumbine_1981       CT  0  1  6 17
2          Walsh_1981       CT 12  3  3  7
3        Brenner_1982       CT  4  1  2 13
4     Villasanta_1983       CT 10  4  3 25
5 vanEngelshoven_1984       CT  3  1  4 12
6          Bandy_1985       CT  9  3  3 29
```

There are two obvious ways to analyze this data set. First, analyze the meta-analysis of each imaging technology separately, which gives each study its own estimates of the hyper-parameters. Second, analyze all studies together and incorporate the stratification using a technology-specific intercept.

To analyze all subdata separately, we call the function `meta4diag()` three times assuming for each subset model (1) without covariate information. Here, we use the default settings of `meta4diag()`.

```
R> res.CT <- meta4diag(data = Scheidler[Scheidler$modality == "CT", ])
R> res.LAG <- meta4diag(data = Scheidler[Scheidler$modality == "LAG", ])
R> res.MRI <- meta4diag(data = Scheidler[Scheidler$modality == "MRI", ])
```

Prior distributions as well as other model details, such as the link function, can be changed as described in Section 3.3.

To plot the results of all three analyses in one device, we can use the `SROC()` function with the argument `add = TRUE`, see Figure 7a.

```
R> SROC(res.CT, dataShow = "o", lineShow = TRUE, prShow = FALSE,
+    data.col = "red", cr.col = "red", sp.col = "red")
R> SROC(res.LAG, dataShow = "o", lineShow = TRUE, prShow = FALSE,
+    data.col = "blue", cr.col = "blue", sp.col = "blue", add = TRUE)
R> SROC(res.MRI, dataShow = "o", lineShow = TRUE, prShow = FALSE,
+    data.col = "green", cr.col = "green", sp.col = "green", add = TRUE)
```

To analyze the entire data set, we consider the column `modality` as a categorical covariate

and use the following model where the overall intercept is omitted:

$$
\begin{aligned}
&\mathrm{TP}_i|\mathrm{Se}_i \sim \mathrm{Binomial}(\mathrm{TP}_i + \mathrm{FN}_i, \mathrm{Se}_i), \quad \mathrm{logit}(\mathrm{Se}_i) = \mu_i + \phi_i, \\
&\mathrm{TN}_i|\mathrm{Sp}_i \sim \mathrm{Binomial}(\mathrm{TN}_i + \mathrm{FP}_i, \mathrm{Sp}_i), \quad \mathrm{logit}(\mathrm{Sp}_i) = \nu_i + \psi_i, \\
&\mu_i = \begin{cases} \mu_{\mathrm{CT}} & \text{if} \quad i = 1, \ldots, 17 \\ \mu_{\mathrm{LAG}} & \text{if} \quad i = 18, \ldots, 34 \\ \mu_{\mathrm{MRI}} & \text{if} \quad i = 35, \ldots, 44 \end{cases} \qquad \nu_i = \begin{cases} \nu_{\mathrm{CT}} & \text{if} \quad i = 1, \ldots, 17 \\ \nu_{\mathrm{LAG}} & \text{if} \quad i = 18, \ldots, 34 \\ \nu_{\mathrm{MRI}} & \text{if} \quad i = 35, \ldots, 44 \end{cases} \qquad (2) \\
&\begin{pmatrix} \phi_i \\ \psi_i \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_\phi^2 & \rho\sigma_\phi\sigma_\psi \\ \rho\sigma_\phi\sigma_\psi & \sigma_\psi^2 \end{pmatrix} \right].
\end{aligned}
$$

Here, $i = 1, \ldots, 44$. To analyze this data in **meta4diag**, we call the function `meta4diag()` with argument `modality = "modality"`:

```
R> res <- meta4diag(data = Scheidler, modality = "modality")
R> res

Time used:
 Pre-processing    Running inla Post-processing           Total
     0.67642403      0.37233186      0.09966302     1.14841890

Model:Binomial-Normal Bivariate Model for Se & Sp.
Data contains 44 primary studies.

Data has Modality variable with level 3.
Covariates not contained.

Model using link function logit.

Marginals can be plotted with setting variable names to
mu.CT, mu.LAG, mu.MRI, nu.CT, nu.LAG, nu.MRI, var1, var2 and rho.
```

To print the estimates for the parameters of the model, we use the function `summary()`:

```
R> summary(res)

Time used:
 Pre-processing    Running inla Post-processing           Total
      0.6840582       0.4051738       0.1005492       1.1897812

Fixed effects:
         mean    sd 0.025quant 0.5quant 0.975quant
mu.CT  -0.144 0.272     -0.689   -0.141      0.386
mu.LAG  0.809 0.263      0.299    0.806      1.340
mu.MRI  0.192 0.347     -0.496    0.193      0.877
nu.CT   2.699 0.270      2.184    2.693      3.249
nu.LAG  1.589 0.231      1.141    1.585      2.057
```

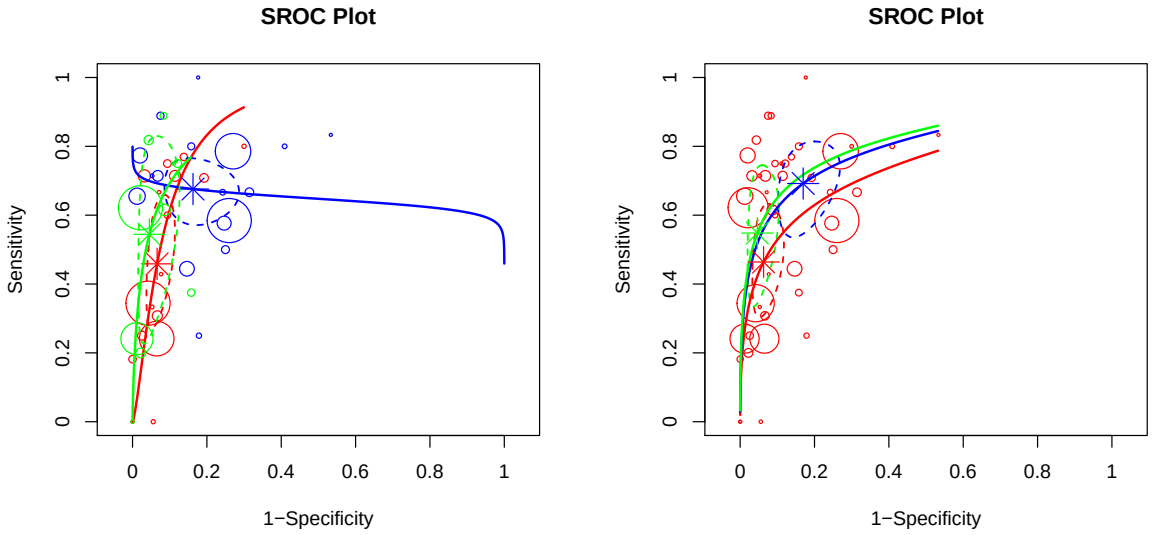
```
nu.MRI  3.027 0.343      2.368    3.021      3.722

Model hyperpar:
           mean    sd 0.025quant 0.5quant 0.975quant
var_phi   0.800 0.308      0.354    0.747      1.550
var_psi   0.701 0.258      0.324    0.658      1.327
cor      -0.481 0.190     -0.790   -0.500     -0.058

-------------------
              mean    sd 0.025quant 0.5quant 0.975quant
mean(Se.CT)  0.465 0.061      0.346    0.465      0.583
mean(Se.LAG) 0.690 0.051      0.586    0.691      0.783
mean(Se.MRI) 0.547 0.077      0.394    0.548      0.692
mean(Sp.CT)  0.935 0.015      0.903    0.937      0.960
mean(Sp.LAG) 0.828 0.030      0.766    0.830      0.882
mean(Sp.MRI) 0.952 0.014      0.918    0.954      0.975

-------------------
Correlation between mu.CT and nu.CT is -0.3013.
Correlation between mu.LAG and nu.LAG is -0.3494.
Correlation between mu.MRI and nu.MRI is -0.3081.

Marginal log-likelihood: -249.72
Variable names for marginal plotting:
      mu.CT, mu.LAG, mu.MRI, nu.CT, nu.LAG, nu.MRI, var1, var2, rho
```

We apply the `SROC()` function again to check the difference between a separate and joint analysis:

```
R> SROC(res, dataShow = "o", lineShow = TRUE, prShow = FALSE,
+    cr.col = c("red", "blue", "green"), sp.col = c("red", "blue", "green"),
+    line.col = c("red", "blue", "green"))
```

Of note is that the SROC curves strongly vary depending on which formula is used to compute them, see Chappell *et al.* (2009) for a discussion. Five different formulas are available in `meta4diag` which can be chosen using the argument `sroc.type`, see Section 3.3 and documentation.

From Figures 7a and 7b, we can see that the estimated summary points are almost the same in both analyses. However, the credible regions change slightly using the different model formulations. More striking are the changes in the SROC curves, in particular for the LAG subset (blue). Looking at the data there is no obvious trend that sensitivity increases along with increasing 1 − specificity. The estimated posterior correlation $\hat{\rho}$ is 0.1809 $[-0.55, 0.79]$. Chappell *et al.* (2009) stated that it is not appropriate to use SROC curves when $\hat{\rho}$ is close to zero or positive. Using separate analyses, we assume that each subdata has its own random effect properties. While using the full data set with a categorical covariate, we assume that all the subdata share the same covariance matrix. The choice of how to model the data is up to the user. However, when the argument `covariates` is used in the modeling, i.e.,



(a) (b)

Figure 7: (a) SROC plot to compare the results for each sub-analysis of the `Scheidler` data set. Red: result for CT data. Blue: result for LAG data. Green: result for MRI data. Bubbles represent the observed values where the size is proportional to the number of study participants, dashed lines are 95% credible regions and the star points are the summary points. The lines show the corresponding SROC curves. (b) SROC plot for the joint analysis of the `Scheidler` data set. Bubbles are the observations, dashed lines are 95% credible regions and the star points are the summary points. The lines show the corresponding SROC curves. Red: result for CT data. Blue: result for LAG data. Green: result for MRI data.

continuous covariates are included, the overall summary points, the confidence region and the prediction region are no longer available through the function `SROC()`, and only the study specific summary points can be obtained instead, see the example in Section 3.5.

The corresponding forest plot for this data set is shown in Figure 8. The plot is automatically separated into three parts due to the column `modality` with three different levels.

```
R> forest(res, accuracy.type = "sens")
```

3.5. Use of continuous covariate information

The Catheter Segment Culture data consists of 33 studies from a diagnostic accuracy analysis by Chu *et al.* (2010). The studies analyzed semi-quantitative (19 studies) and quantitative (14 studies) catheter segment culture for the diagnosis of intravascular device-related blood stream infection. In the data set a column with name `type` indicates whether a study is based on semi-quantitative or quantitative catheter segment culture. We consider `type` as a categorical covariate in the model so that it should be set to `modality = "type"`. We choose this data set as an example because it contains an additional column with name `prevalence` providing disease prevalence information which can be considered as a continuous covariate. To analyze the data set, we first load the `Catheter` data set:

```
R> data("Catheter", package = "meta4diag")
```

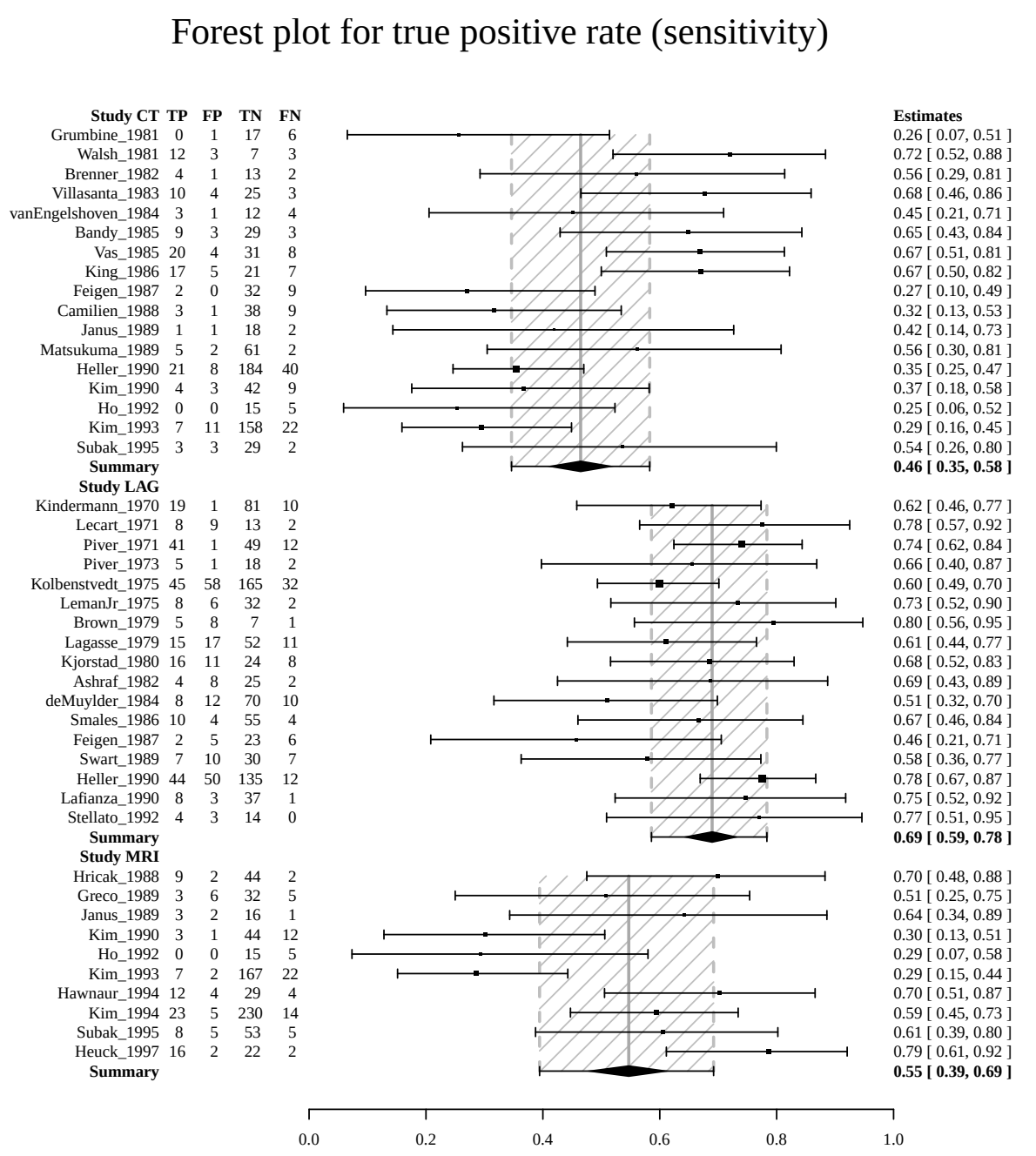


Figure 8: Forest plot for the `Scheidler` data. The plot is separated into three parts relating to the three sub-data sets.

```
R> head(Catheter)
```

```
      studynames                type prevalence TP  FP  TN FN
1    Cooper_1985 Semi-quantitative        3.6 12  29 289  0
2 Gutierrez_1992 Semi-quantitative       12.2 10  14  72  2
3 Cercenado_1990 Semi-quantitative       12.9 17  36  85  1
```

```
4     Rello_1991 Semi-quantitative       13.2 13  18  67  0
5      Maki_1977 Semi-quantitative        1.6  4  21 225  0
6 Aufwerber_1991 Semi-quantitative        3.1 15 122 403  2
```

Consider that we would like to use the model:

$$
\begin{aligned}
&\mathrm{FN}_i|\mathrm{Se}_i \sim \mathrm{Binomial}(\mathrm{TP}_i + \mathrm{FN}_i, \mathrm{Se}_i), && \mathrm{logit}(\mathrm{Se}_i) = \mu_i + \alpha \cdot \mathrm{prevalence}_i + \phi_i,\\
&\mathrm{TP}_i|1-\mathrm{Sp}_i \sim \mathrm{Binomial}(\mathrm{TN}_i + \mathrm{FP}_i, 1-\mathrm{Sp}_i), && \mathrm{logit}(1-\mathrm{Sp}_i) = \nu_i + \beta \cdot \mathrm{prevalence}_i + \psi_i,\\
&\mu_i = \begin{cases} \mu_{\text{semi-quantitative}} & \text{if} \quad i = 1,\ldots,19 \\ \mu_{\text{quantitative}} & \text{if} \quad i = 20,\ldots,33 \end{cases} && \nu_i = \begin{cases} \nu_{\text{semi-quantitative}} & \text{if} \quad i = 1,\ldots,19 \\ \nu_{\text{quantitative}} & \text{if} \quad i = 20,\ldots,33 \end{cases}\\
&\begin{pmatrix} \phi_i \\ \psi_i \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_\phi^2 & \rho\sigma_\phi\sigma_\psi \\ \rho\sigma_\phi\sigma_\psi & \sigma_\psi^2 \end{pmatrix} \right].
\end{aligned}
\tag{3}
$$

That means we would like to model sensitivity and 1 − specificity jointly as proposed by Chu *et al.* (2010) for this data set. This can be done by setting `model.type = 2`. As the Catheter Segment Culture data contains one categorical covariate `type` and one continuous covariate `prevalence`, the argument `modality` is set to be `"type"` and argument `covariates` is set to be `"prevalence"`.

```
R> set.seed(19876)
R> res <- meta4diag(data = Catheter, model.type = 2, var.prior = "PC",
+    var2.prior = "PC", cor.prior = "PC", var.par = c(3, 0.05),
+    cor.par = c(1, -0.1, 0.5, -0.95, 0.05, NA, NA),
+    modality = "type", covariates = "prevalence",
+    quantiles = c(0.125, 0.875), nsample = 10000, seed = 1352)
```

Currently only one categorical covariate can be included in the model, whereas there is no limitation for the number of continuous covariates. In order to include more than one continuous covariate in the model, the user can provide a vector giving the names of all covariates to be included or the respective column numbers in the data frame.

Here, we choose a PC prior for all hyperparameters. The vector of parameters for the PC prior of the correlation parameter must always be of length 7 specifying as `c(strategy`, ρ_0, ω, u_1, α_1, u_2, α_2`)`. However, u_2 and α_2 are not required when using `strategy = 1`, u_1 and α_1 are not required when `strategy = 2` and there is no need to specify ω when `strategy = 3`, see Section 2.2. To obtain the 12.5% and 87.5% quantiles in addition to the default 2.5%, 50% and 97.5% quantiles we set `quantiles = c(0.125, 0.875)`. Summary estimates are again obtained using the function `summary()`:

```
R> summary(res)
```

```
Time used:
 Pre-processing    Running inla Post-processing           Total
      0.9602380       0.6268680       0.1075032       1.6946092

Fixed effects:
                       mean    sd 0.025quant 0.125quant 0.5quant 0.875quant
```

```
mu.Semi.quantitative  1.827 0.360      1.170      1.428    1.808      2.240
mu.Quantitative       1.758 0.439      0.940      1.269    1.739      2.259
nu.Semi.quantitative -1.999 0.236     -2.468     -2.267   -1.998     -1.732
nu.Quantitative      -2.690 0.317     -3.326     -3.050   -2.686     -2.332
alpha.prevalence      0.006 0.015     -0.024     -0.011    0.006      0.022
beta.prevalence       0.032 0.012      0.008      0.019    0.032      0.046
                     0.975quant
mu.Semi.quantitative      2.594
mu.Quantitative           2.681
nu.Semi.quantitative     -1.535
nu.Quantitative          -2.074
alpha.prevalence          0.034
beta.prevalence           0.057

Model hyperpar:
          mean    sd 0.025quant 0.125quant 0.5quant 0.875quant 0.975quant
var_phi  1.039 0.481      0.391      0.561    0.942      1.575      2.249
var_psi  0.764 0.232      0.416      0.521    0.727      1.031      1.322
cor      0.094 0.216     -0.327     -0.161    0.094      0.349      0.506

Marginal log-likelihood: -239.5213
Variable names for marginal plotting:
      mu.Semi.quantitative, mu.Quantitative, nu.Semi.quantitative,
      nu.Quantitative, alpha.prevalence, beta.prevalence, var1, var2, rho
```

A forest plot for the log diagnostic odds ratio is given in Figure 9. Here, 75% credible intervals are shown which is specified by setting the argument `intervals = c(0.125, 0.875)` within the function `forest()`.

```
R> forest(res, accuracy.type = "LDOR", est.type = "median", nameShow = TRUE,
+    estShow = "left", dataShow = "center", text.cex = 1.5, arrow.lwd = 1.5,
+    cut = c(0, 10), intervals = c(0.125, 0.875))
```

Of note is that when the argument `covariates` is available, the summary estimates cannot be returned through the function `forest()`. Similarly, the summary points, confidence region and prediction region in the SROC plot are not available. The SROC curve in contrast is still available. However, it does not depend on the choice of the argument `sroc.type`, but is computed according to Walter (2002) by fitting a regression equation

$$D_i = a + bS_i,$$

where

$$D_i = \log\left(\frac{\widehat{\mathrm{Se}}_i}{1-\widehat{\mathrm{Se}}_i}\right) - \log\left(\frac{1-\widehat{\mathrm{Sp}}_i}{\widehat{\mathrm{Sp}}_i}\right)$$

and

$$S_i = \log\left(\frac{\widehat{\mathrm{Se}}_i}{1-\widehat{\mathrm{Se}}_i}\right) + \log\left(\frac{1-\widehat{\mathrm{Sp}}_i}{\widehat{\mathrm{Sp}}_i}\right)$$

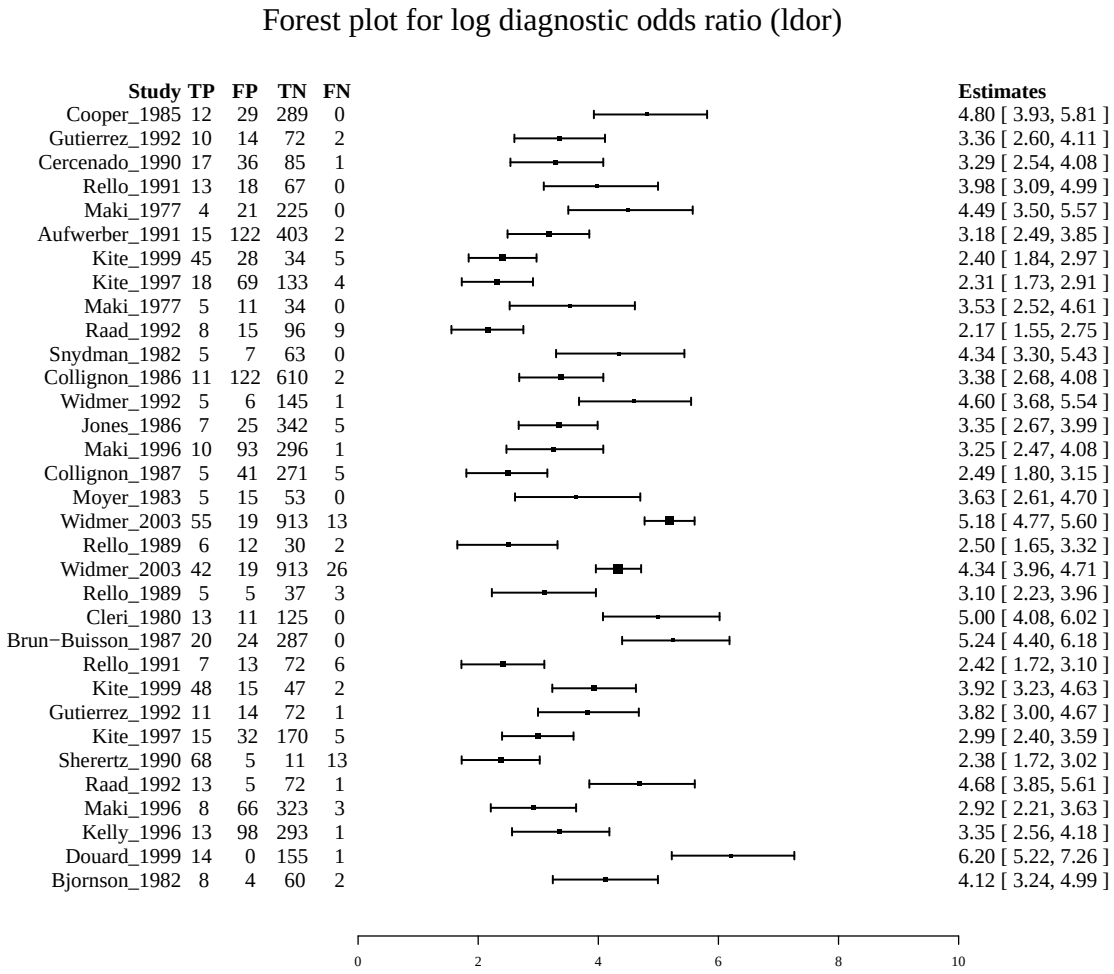


Figure 9: Forest plot for the log diagnostic odds ratio (LDOR) of the `Catheter` data set. The study names, original data set, estimated mean and corresponding 75% credible intervals are also shown.

respectively. After fitting the regression line, the equation of the SROC curve can be obtained as

$$\mathrm{SROC}(x) = \frac{\exp\left(\frac{a}{1-b}\right) x^{(1+b)/(1-b)}}{1 + \exp\left(\frac{a}{1-b}\right) x^{(1+b)/(1-b)}}, \quad x \in [0, 1].$$

3.6. Graphical user interface

To make Bayesian diagnostic meta-analysis easier to use for applied scientists, a cross-platform, interactive and user-friendly graphical user interface has been implemented. The graphical user interface can be used to load the data, set and graphically inspect the priors as the hyperparameters are manually changed by sliders (see Figure 12), and run the model. The results of the analysis are shown directly in the interface and can be saved for later use. The graphical user interface only requires the basic knowledge of R required to start R, load the packages and run the command that starts the graphical user interface. Within the

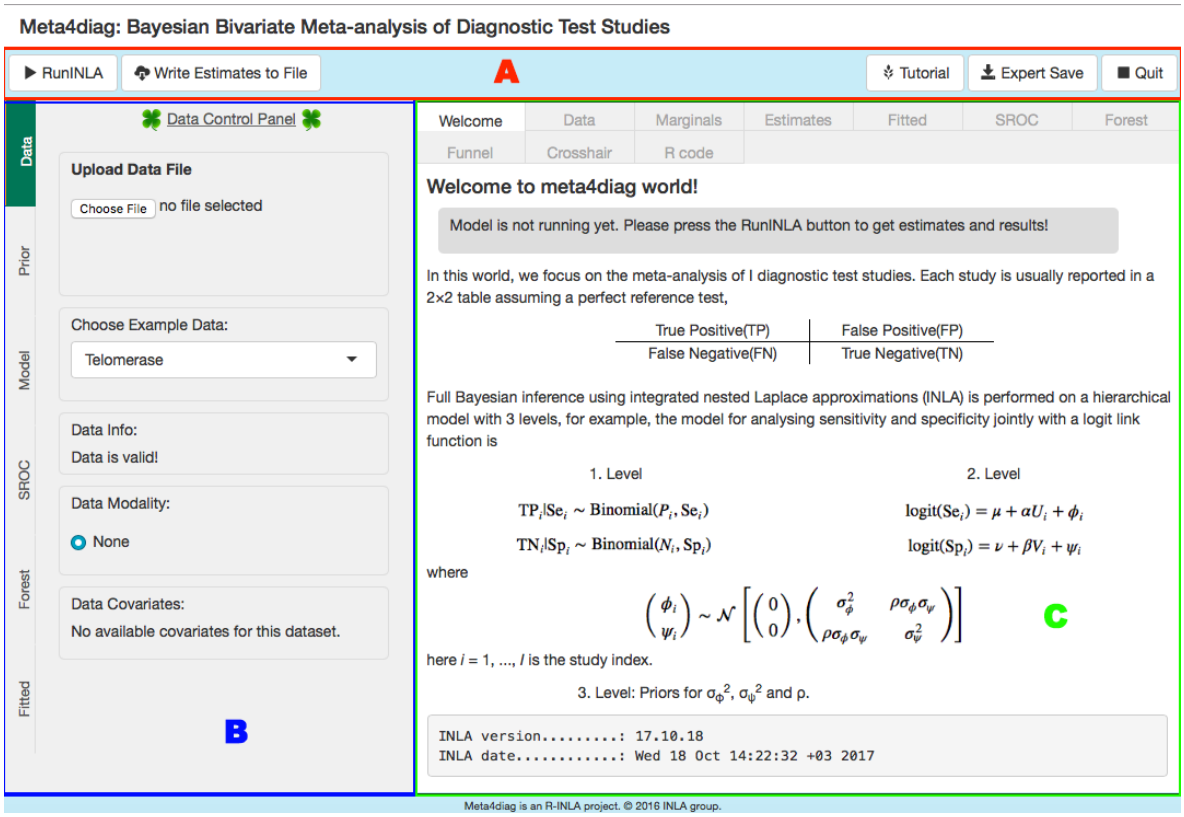


Figure 10: GUI main window of **meta4diag** after start up. (A) toolbar, (B) tool control panel, (C) view area, showing different pages (welcome message, data set, summary results, graphics). In particular, the "Data Control Panel" is shown in the "tool control panel" area. Users could upload their own data sets for analyzing or choose an example data set for understanding the package. The "Welcome" page is shown in the "view area". The basic information for modeling and the description of bivariate meta-analysis of diagnostic test studies are shown in this page.

interface all options are visualized as buttons or drop-down menus, and help for each option is found as tooltips when the user moves the mouse over the option or the "Description area". The interface has been tested in the browsers "Internet Explorer", "Mozilla Firefox", "Google Chrome" and "Safari" on Linux, Mac and Windows 10 operating systems.

The graphical user interface is started by loading the packages **meta4diag** and **INLA** and then calling the function `meta4diagGUI()` with:

```
R> library("meta4diag")
R> library("INLA")
R> meta4diagGUI()
```

The start window of the graphical user interface is shown in Figure 10 and is divided into three areas A, B and C. A contains the toolbar and has buttons for running **INLA** and writing the results to a text file, and buttons for starting the tutorial, saving the results to an R object for further study in R and for quitting the interface. B has 6 tabs that contain

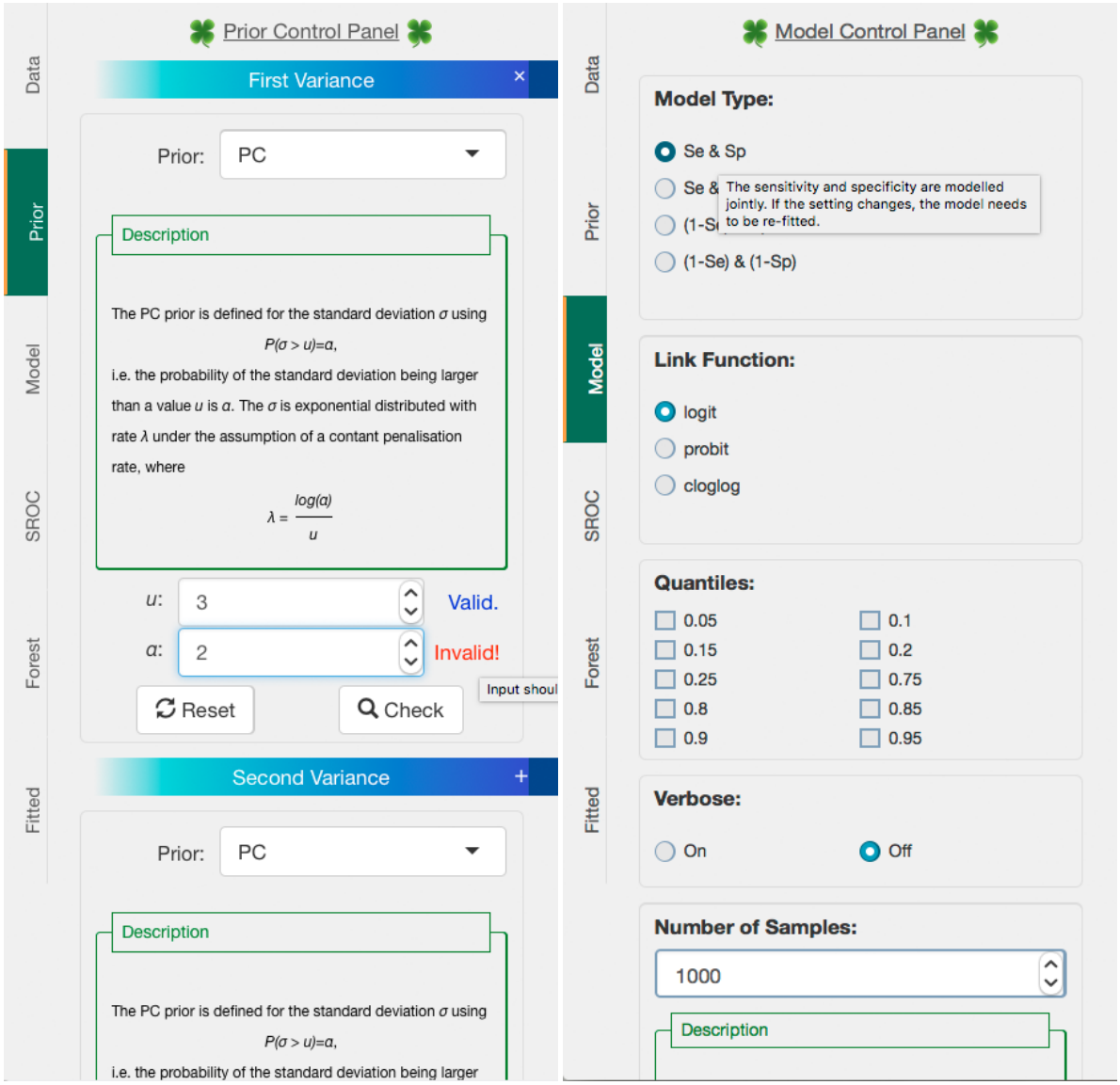


Figure 11: Details for two tool panels. Left: "Prior Control Panel". In this panel, users can set the prior distributions for the first variance component, the second variance component and the correlation. In particular, the specification of the PC prior for the first variance component is shown. The "Description area" is shown to explain what the prior is. The red "Invalid!" indicates that the given value for the hyperparameter α is not valid. The interval of the valid values can be seen from the tooltips of the indicator "Invalid". Right: "Model Control Panel".

the various control panels, which are used to set up the analysis, such as the data control panel, the prior control panel, and the model control panel. The options within these three panels must be set before pressing the "RunINLA" button in *A*. The "Forest" control panel and "SROC" control panel in *B* are used for choosing plotting settings, and can be used both before and after running the model, and the "Fitted" panel allows the user to inspect the estimates for different choices of accuracy types and can also be set after running the model.

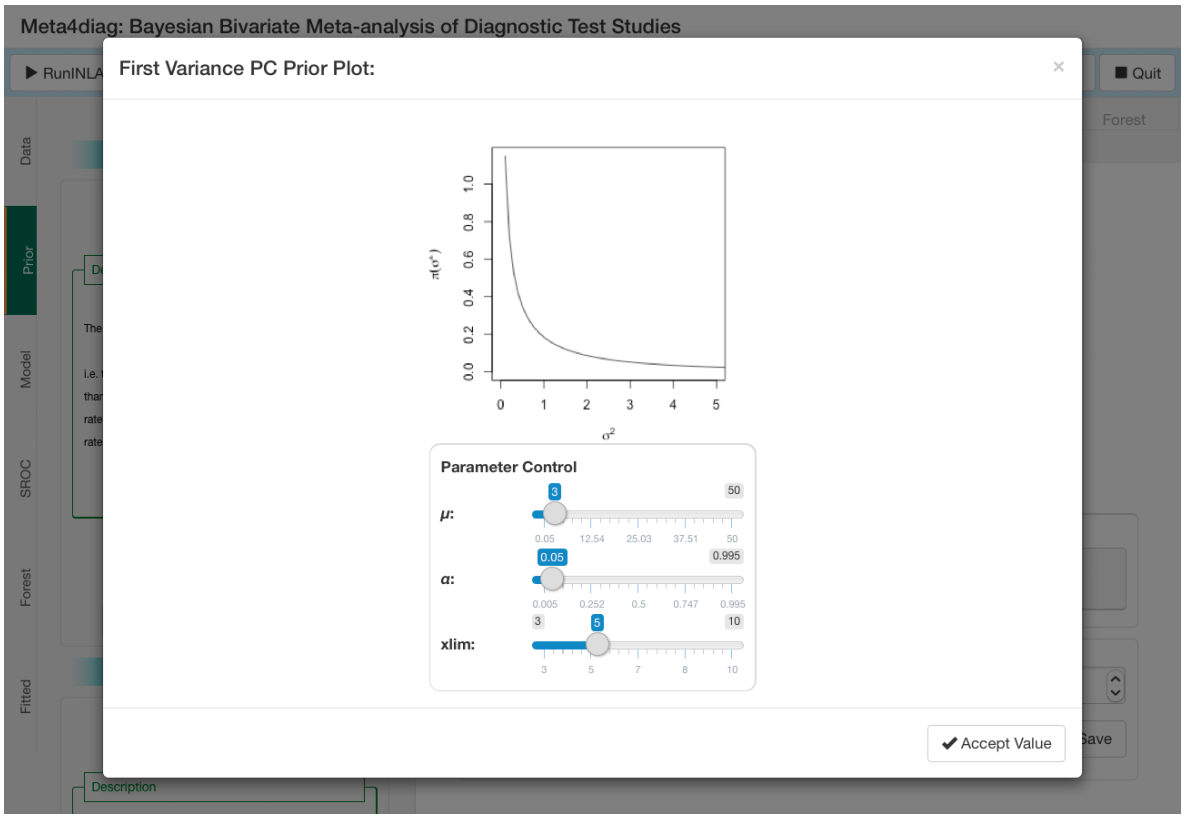


Figure 12: Interactive prior specification window. The prior density is shown and can be changed by sliding the bottom bars.

Lastly, C has 10 tabs where the first is a welcome page and the rest are used to view the data and the results.

Figure 11, which contains a screenshot of the "Prior Control Panel" on the left-hand side and the "Model Control Panel" on the right-hand side, gives an example of how the user can set the model and the prior. The description of the options in each panel is integrated in the GUI through tooltips, but can also be found in the package documentation (see the man page for `meta4diag()` for details). The left-hand side of the screenshot only shows how the user can set the prior distribution for the first variance component, but the panel also contains options for setting the priors on the second variance component and the correlation in the bivariate model. Figure 12 illustrates how the user can explore different settings of the hyperparameters interactively by sliding the sliders corresponding to each parameter. When the PC prior is selected for the correlation parameter, the user may use either of the specification strategies described in Section 2.2 to set the hyperparameters. The "Model Control Panel" shown on the right-hand side of Figure 11 is used to specify the model type, link function, quantiles of interest, and more.

After setting the options in the first three control panels and clicking the "RunINLA" button, the chosen data set will be loaded and analyzed. The results of the analysis will be shown in the view area (C) and, for example, the SROC plot can be viewed in the SROC tab of the view area (C) as shown in Figure 13. The other tabs can be used to view summary estimates, study-specific accuracy estimates, posterior marginal plots and the forest plot. In each case

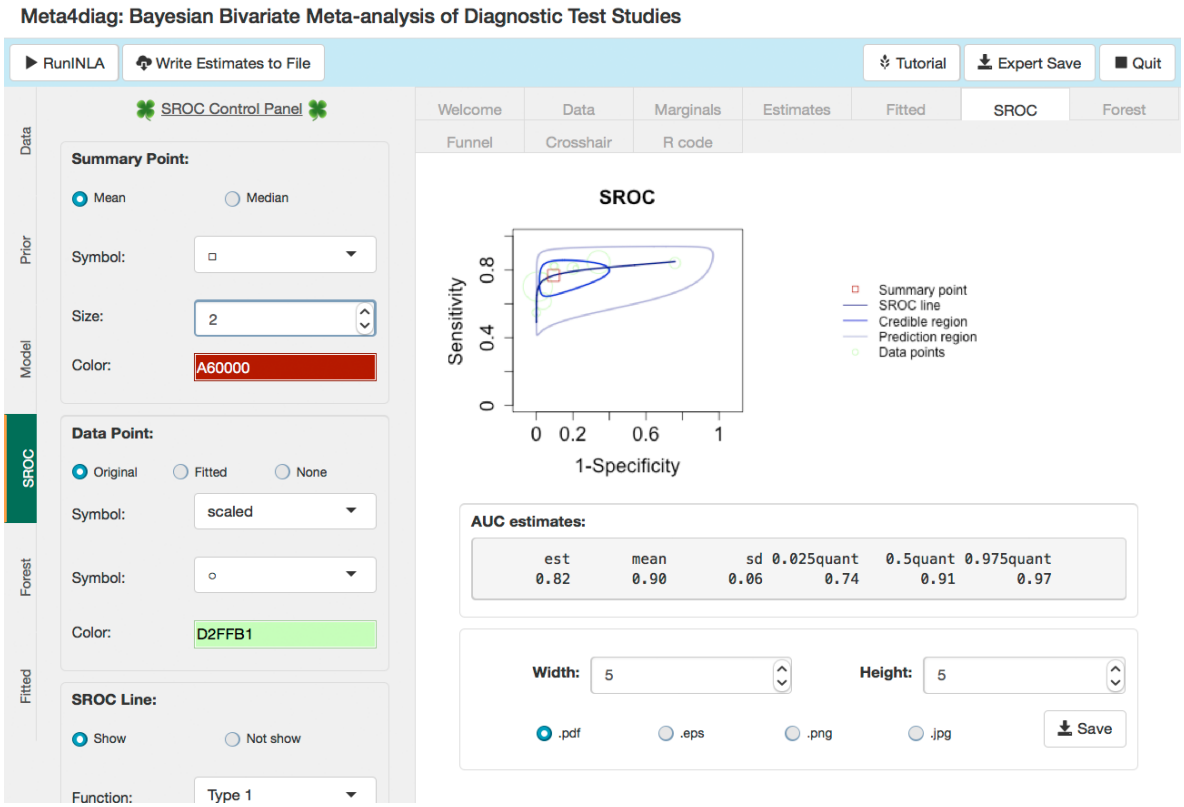


Figure 13: Example of an SROC plot in the view area and the "SROC Control Panel" in the left tool bar.

the R code that generated the figure or text is also shown. If the data, the model or prior settings are changed, the user must push the "RunINLA" button again to update the results.

4. Conclusion

The present paper demonstrates the usage of the R package **meta4diag** for analyzing bivariate meta-analyses of diagnostic test studies with R, and illustrates its usage using three examples from the literature. The package is built on top of the R package **INLA** and thus provides full Bayesian inference without the need for Markov chain Monte Carlo techniques. This is especially important when several or complex meta-analyses are studied, or simulation studies shall be performed, as then the time speed-up becomes obvious. The model can be easily specified, whereby the user does not need to know any **INLA**-specific details. Quantities relevant in the field of diagnostic meta-regression are internally computed and returned directly without requiring the user to work with the general and complex **INLA** output.

One of the biggest advantages – besides of being fast compared to other software packages for Bayesian inference – is the flexible and at the same time intuitive prior specification framework. In particular the newly proposed PC priors (Simpson *et al.* 2017) are supported. Here, the user has the possibility to incorporate expert knowledge in the form of probability

contrasts. Guo *et al.* (2017) compared the performance of different PC priors with previously proposed priors in the bivariate model through an intensive simulation study and a real data set. Both informative and less informative PC priors were studied, and results indicated that the PC priors perform at least as good as previously used priors.

A graphical user interface makes the package also attractive for users who prefer to work with interactive windows offering selection menus. The graphical user interface provides the full functionality of the package. In addition the user can inspect the priors directly and change them interactively. By offering fast inference within a Bayesian framework, intuitive choice of prior distributions and the graphical user interface we feel that this package has great potential for routine practice. As a future research direction, we would like to expand the functionality of this package to a three-variate model analyzing sensitivity, specificity and disease prevalence jointly. Further, we would like to investigate how to extend package **meta4diag** when the assumption of a perfect reference test is not given.

Acknowledgments

We are grateful for helpful discussions with Håvard Rue and Geir-Arne Fuglstad.

References

Altman DG (1990). *Practical Statistics for Medical Research*. CRC Press.

Arends LR, Hamza TH, Houwelingen JCV, Heijenbrok-Kal MH, Hunink MG, Stijnen T (2008). "Bivariate Random Effects Meta-Analysis of ROC Curves." *Medical Decision Making*, **28**(5), 621–638. doi:10.1177/0272989x08319957.

Borenstein M, Hedges LV, Higgins JPT, Rothstein HR (2011). *Introduction to Meta-Analysis*. John Wiley & Sons.

Chang W, Cheng J, Allaire JJ, Xie Y, McPherson J (2017). ***shiny**: Web Application Framework for R*. R package version 1.0.5, URL https://CRAN.R-project.org/package=shiny.

Chappell FM, Raab GM, Wardlaw JM (2009). "When Are Summary ROC Curves Appropriate for Diagnostic Meta-Analyses?" *Statistics in Medicine*, **28**(21), 2653–2668. doi:10.1002/sim.3631.

Chen DGD, Peace KE (2013). *Applied Meta-Analysis with R*. CRC Press, Boca Raton.

Chen Y, Chu H, Luo S, Nie L, Chen S (2015). "Bayesian Analysis on Meta-Analysis of Case-Control Studies Accounting for Within-Study Correlation." *Statistical Methods in Medical Research*, **24**(6), 836–855. doi:10.1177/0962280211430889.

Chu H, Cole SR (2006). "Bivariate Meta-Analysis of Sensitivity and Specificity with Sparse Data: A Generalized Linear Mixed Model Approach." *Journal of Clinical Epidemiology*, **59**(12), 1331–1332. doi:10.1016/j.jclinepi.2006.06.011.

Chu H, Guo H, Zhou Y (2010). "Bivariate Random Effects Meta-Analysis of Diagnostic Studies Using Generalized Linear Mixed Models." *Medical Decision Making*, **30**(4), 499–508. doi:10.1177/0272989x09353452.

Diaz M (2015). "Performance Measures of the Bivariate Random Effects Model for Meta-Analyses of Diagnostic Accuracy." *Computational Statistics & Data Analysis*, **83**, 82–90. doi:10.1016/j.csda.2014.09.021.

Doebler P (2017). "**mada**: Meta-Analysis of Diagnostic Accuracy." R package version 0.5.8, URL https://CRAN.R-project.org/package=mada.

Gasparrini A, Armstrong B, Kenward MG (2012). "Multivariate Meta-Analysis for Non-Linear and Other Multi-Parameter Associations." *Statistics in Medicine*, **31**(29), 3821–3839. doi:10.1002/sim.5471.

Glas AS, Roos D, Deutekom M, Zwinderman AH, Bossuyt PMM, Kurth KH (2003). "Tumor Markers in the Diagnosis of Primary Bladder Cancer. A Systematic Review." *The Journal of Urology*, **169**(6), 1975–1982. doi:10.1097/01.ju.0000067461.30468.6d.

Guo J, Riebler A (2017). ***meta4diag**: Meta-Analysis for Diagnostic Test Studies*. R package version 2.0.7, URL https://CRAN.R-project.org/package=meta4diag.

Guo J, Riebler A, Rue H (2017). "Bayesian Bivariate Meta-Analysis of Diagnostic Test Studies with Interpretable Priors." *Statistics in Medicine*, **36**(19), 3039–3058. doi:10.1002/sim.7313.

Hamza TH, Arends LR, Van Houwelingen HC, Stijnen T (2009). "Multivariate Random Effects Meta-Analysis of Diagnostic Tests with Multiple Thresholds." *BMC Medical Research Methodology*, **9**(73), 1–15. doi:10.1186/1471-2288-9-73.

Hamza TH, Reitsma JB, Stijnen T (2008). "Meta-Analysis of Diagnostic Studies: A Comparison of Random Intercept, Normal-Normal, and Binomial-Normal Bivariate Summary ROC Approaches." *Medical Decision Making*, **28**(5), 639–649. doi:10.1177/0272989x08323917.

Harbord RM (2011). "Commentary on "Multivariate Meta-Analysis: Potential and Promise"." *Statistics in Medicine*, **30**(20), 2507–2508. doi:10.1002/sim.4278.

Harbord RM, Deeks JJ, Egger M, Whiting P, Sterne JAC (2007). "A Unification of Models for Meta-Analysis of Diagnostic Accuracy Studies." *Biostatistics*, **8**(2), 239–251. doi:10.1093/biostatistics/kxl004.

Harbord RM, Whiting P (2009). "**metandi**: Meta-Analysis of Diagnostic Accuracy Using Hierarchical Logistic Regression." *Stata Journal*, **9**(2), 211–229.

Kuss O, Hoyer A, Solms A (2014). "Meta-Analysis for Diagnostic Accuracy Studies: A New Statistical Model Using Beta-Binomial Distributions and Bivariate Copulas." *Statistics in Medicine*, **33**(1), 17–30. doi:10.1002/sim.5909.

Lewis S, Clarke M (2001). "Forest Plots: Trying to See the Wood and the Trees." *BMJ: British Medical Journal*, **322**, 1479–1480. doi:10.1136/bmj.322.7300.1479.

Lindgren F, Rue H (2015). "Bayesian Spatial Modelling with R-**INLA**." *Journal of Statistical Software*, **63**(19), 1–25. doi:10.18637/jss.v063.i19.

Lunn D, Spiegelhalter D, Thomas A, Best N (2009). "The BUGS Project: Evolution, Critique and Future Directions." *Statistics in Medicine*, **28**(25), 3049–3067. doi:10.1002/sim.3680.

Lunn DJ, Thomas A, Best N, Spiegelhalter D (2000). "**WinBUGS** – A Bayesian Modelling Framework: Concepts, Structure, and Extensibility." *Statistics and Computing*, **10**(4), 325–337. doi:10.1023/a:1008929526011.

Menke J (2014). "Bayesian Bivariate Meta-Analysis of Sensitivity and Specificity: Summary of Quantitative Findings in 50 Meta-Analyses." *Journal of Evaluation in Clinical Practice*, **20**(6), 844–852. doi:10.1111/jep.12173.

Paul M, Riebler A, Bachmann LM, Rue H, Held L (2010). "Bayesian Bivariate Meta-Analysis of Diagnostic Test Studies Using Integrated Nested Laplace Approximations." *Statistics in Medicine*, **29**(12), 1325–1339. doi:10.1002/sim.3858.

Phillips B, Stewart LA, Sutton AJ (2010). "'Cross Hairs' Plots for Diagnostic Meta-Analysis." *Research Synthesis Methods*, **1**(3–4), 308–315. doi:10.1002/jrsm.26.

Plummer M (2003). "**JAGS**: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling." In K Hornik, F Leisch, A Zeileis (eds.), *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)*. Technische Universität Wien, Vienna, Austria. URL https://www.R-project.org/conferences/DSC-2003/Proceedings/Plummer.pdf.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Reitsma JB, Glas AS, Rutjes AWS, Scholten RJPM, Bossuyt PM, Zwinderman AH (2005). "Bivariate Analysis of Sensitivity and Specificity Produces Informative Summary Measures in Diagnostic Reviews." *Journal of Clinical Epidemiology*, **58**(10), 982–990. doi:10.1016/j.jclinepi.2005.02.022.

Riley RD, Abrams KR, Sutton AJ, Lambert PC, Thompson JR (2007). "Bivariate Random-Effects Meta-Analysis and the Estimation of Between-Study Correlation." *BMC Medical Research Methodology*, **7**(3), 1–15. doi:10.1186/1471-2288-7-3.

Rue H, Martino S, Chopin N (2009). "Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations." *Journal of the Royal Statistical Society B*, **71**(2), 319–392. doi:10.1111/j.1467-9868.2008.00700.x.

Rutter CM, Gatsonis CA (2001). "A Hierarchical Regression Approach to Meta-Analysis of Diagnostic Test Accuracy Evaluations." *Statistics in Medicine*, **20**(19), 2865–2884. doi:10.1002/sim.942.

SAS Institute Inc (2013). *SAS/STAT Software, Version 13.2*. SAS Institute Inc., Cary. URL http://www.sas.com/.

Scheidler J, Hricak H, Yu KK, Subak L, Segal MR (1997). "Radiological Evaluation of Lymph Node Metastases in Patients with Cervical Cancer: A Meta-Analysis." *The Journal of the American Medical Association*, **278**(13), 1096–1101.

Schiller I, Dendukuri N (2015). ***HSROC**: Joint Meta-Analysis of Diagnostic Test Sensitivity and Specificity With or Without a Gold Standard Reference Test*. R package version 2.1.8, URL https://CRAN.R-project.org/package=HSROC.

Simpson D, Rue H, Riebler A, Martins TG, Sørbye SH (2017). "Penalising Model Component Complexity: A Principled, Practical Approach to Constructing Priors." *Statistical Science*, **32**(1), 1–28. doi:10.1214/16-sts576.

StataCorp (2015). *Stata Data Analysis Statistical Software: Release 14*. StataCorp LP, College Station. URL http://www.stata.com/.

Takwoingi Y, Deeks J (2008). "**METADAS**: A SAS Macro for Meta-Analysis of Diagnostic Accuracy Studies." In *Methods for Evaluating Medical Tests Symposium. University of Birmingham*.

Van Houwelingen HC, Arends LR, Stijnen T (2002). "Advanced Methods in Meta-Analysis: Multivariate Approach and Meta-Regression." *Statistics in Medicine*, **21**(4), 589–624. doi:10.1002/sim.1040.

Verde PE (2010). "Meta-Analysis of Diagnostic Test Data: A Bivariate Bayesian Modeling Approach." *Statistics in Medicine*, **29**(30), 3088–3102. doi:10.1002/sim.4055.

Verde PE (2017). ***bamdit**: Bayesian Meta-Analysis of Diagnostic Test Data*. R package version 3.1.0, URL https://CRAN.R-project.org/package=bamdit.

Wakefield J (2007). "Disease Mapping and Spatial Regression with Count Data." *Biostatistics*, **8**(2), 158–183. doi:10.1093/biostatistics/kxl008.

Walter SD (2002). "Properties of the Summary Receiver Operating Characteristic (SROC) Curve for Diagnostic Test Data." *Statistics in Medicine*, **21**(9), 1237–1256. doi:10.1002/sim.1099.

White IR (2009). "Multivariate Random-Effects Meta-Analysis." *Stata Journal*, **9**(1), 40–56.

Youden WJ (1950). "Index for Rating Diagnostic Tests." *Cancer*, **3**(1), 32–35. doi:10.1002/1097-0142(1950)3:1<32::aid-cncr2820030106>3.0.co;2-3.

Affiliation:

Jingyi Guo, Andrea Riebler
Department of Mathematical Sciences
Norwegian University of Science and Technology
7491 Trondheim, Norway
E-mail: jingyi.guo@math.ntnu.no, andrea.riebler@math.ntnu.no

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
January 2018, Volume 83, Issue 1 *Submitted:* 2015-07-02
doi:10.18637/jss.v083.i01 *Accepted:* 2017-02-28