BayesSyneRgy: Flexible Bayesian modelling of synergistic interaction effects in in-vitro drug combination experiments

Andrea Cremaschi[*]
Oslo University Hospital and University of Oslo

Leiv Rønneberg[*]
University of Oslo

Arnoldo Frigessi
University of Oslo

Manuela Zucknick
University of Oslo

Abstract

The study of the effect of cancer therapies is often tested pre-clinically via in-vitro experiments, where the viability of the cell population is measured through cell counts. In this way, large libraries of compounds can be tested, comparing the efficacy of the drugs in fighting the malignancy. A similar approach is used to test combinations of drugs and study their efficacy, as well as their interactions. Drug-drug interaction studies focus on the quantification of the additional effect encountered when two drugs are combined, as opposed to using the treatments separately. In the **BayesSyneRgy** R package, we have implemented a probabilistic model for the description of the drug combination experiment, where the observed dose response curve is fitted as a sum of the expected dose response curve under a zero-interaction model and an additional interaction (synergistic or antagonistic) effect. The interaction surface can be modelled in a flexible manner, either using Bayesian cubic splines or Gaussian processes with several possible kernels. Since the proposed approach is based on a statistical model, it allows us to naturally include replicates of the experiments and to evaluate the uncertainty around the estimates. Posterior estimates of the zero-interaction level and of the interaction term, obtained via adaptive MCMC algorithms, can be used to compute interpretable measures of efficacy of the combined experiment. The implementation of the MCMC sampler is done in C++ for computational speed.

Keywords: Concentration-response study; Drug-drug interaction; Personalized cancer therapy; Proliferation assay (viability); Synergy; Splines; Gaussian Processes; Bayesian modelling.

1. Introduction

In pre-clinical cancer drug sensitivity screening, the effectiveness of compounds is tested in-vitro on cell lines or samples derived from patients. The output is typically measured via concentration-response experiments, where the response of a sample is measured after being exposed to a range of concentrations of the drug for a specified amount of time. The response is measured by cell viability assays or other cell count assays, and the output for a single-drug experiment is usually modelled by fitting a parametric log-logistic model to the concentration-response curve:

$$h(x|m,\lambda) = \left(1 + \left(\frac{x}{m}\right)^{\lambda}\right)^{-1}, \quad x \in \mathbb{R}^+. \tag{1}$$

Positive values of the slope parameter λ are associated with measuring inhibition of cell viability (rather than for example toxicity); this is assumed throughout this work. The inflection point of the curve, m, is interpretable as the *Half-Maximal Effective Concentration* (or EC_{50}), which corresponds to the concentration of the compound needed to induce a response of the sample equal to 50% of the maximum response, a very popular measure of efficacy of the compounds.

In drug combination studies, more than one compound is tested at the same time, with the aim of understanding their interaction. In particular, one is interested in identifying drug combinations that are either *synergistic* or *antagonistic*. An interaction between the drugs that results in a combined effect, which is greater than the total of the individual effects of each drug, is called synergistic, while a diminished effect in the combination is called antagonistic. The output of drug combination experiments can be modelled using suitable mathematical models with the aim of quantifying the interaction component. Building on a long tradition of work starting in the first half of the twentieth century (e.g. Loewe and Muischnek 1926; Bliss 1939), the literature on the topic has developed widely in recent years (see Greco, Bravo, and Parsons 1995; Fouquier and Guedj 2015; Meyer, Wooten, Paudel, Bauer, Hardeman, Westover, Lovly, Harris, Tyson, and Quaranta 2019, for reviews on the topic).

There already exist several software options for analysing drug sensitivity data. Notably, the R packages **synergyfinder** for drug combination studies (He, Kulesskiy, Saarela, Turunen, Wennerberg, Aittokallio, and Tang 2018), and **drc** for single drug data (Ritz, Baty, Streibig, and Gerhard 2015). In addition, standalone software such as **Combenefit** (Di Veroli, Fornari, Wang, Mollard, Bramhall, Richards, and Jodrell 2016) has been utilised to quantify interaction effects in high-throughput drug combination experiments, where thousands of experiments are analysed simultaneously.

The **synergyfinder** package and online web application (https://synergyfinder.fimm.fi/) provides a framework for working with drug combination data, gives summary scores for the overall interaction effect, and allows the user to visualise the estimated response and interaction surfaces. The software allows for several popular choices of the non-interaction model (see section 2), but does not allow for replicate measurements, and cannot provide uncertainty quantification of the final estimate. **Combenefit** allows replicate observations, but likewise does not provide any kind of uncertainty quantification of the interaction estimates. In addition, the drug-interaction landscape can be complex, with some areas exhibiting large synergistic effects, while others highly antagonistic. The existing solutions simply average or integrate over the entire surface and are thus not able to separate these two effects properly, meaning that a highly active surface could have a summary statistic showing little to no overall interaction effect.

Furthermore, a common drawback with all classical synergy models implemented in these packages (see e.g. Yadav, Wennerberg, Aittokallio, and Tang 2015, for an overview), is that these models interpret all deviation of the observed data from the expected values under a specific non-interaction model as interaction, i.e. as evidence for synergistic or antagonistic effects. Therefore, these models do not allow for heterogeneity in the data, for measurement errors or for any other biological or technical variation.

To remedy these problems, in the R package **BayeSyneRgy** we have implemented a statistical model-based approach to the study of interaction between two drugs, where the drug combination surface is modelled using flexible Bayesian approaches. This allows us to take into account all variability in the data in a proper way, and furthermore to use replicate measurements where available to improve estimates. Following Cremaschi, Frigessi, Taskén, and Zucknick (2019), the drug response surface, depending on the two concentrations of the two drugs, is interpreted as the result of a simple stochastic model, which allows to discriminate between its zero-interaction and interaction parts. While the zero-interaction part is modelled parametrically by a product of two log-logistic curves which represent the mono-therapy response curves for each drug individually, the interaction part is modelled non-parametrically in a flexible manner. The user can explore different options for this part of the model, in particular Bayesian splines and several Gaussian process models. At this point, the squared exponential (also called radial basis function or Gaussian kernel), rational quadratic, and Matérn kernels are implemented. We provide several options for typical prior setups, but we have taken care to set good default values for the hyper-prior distributions, which will work well in typical concentration-response data sets.

Before describing the R package in detail and illustrating the use of the software with several examples in Sections 3 to 6, we first introduce the main model in Section 2, where we summarize the Bayesian splines-based model (first introduced in Cremaschi *et al.* 2019), as well as the Gaussian process-based models.

2. Modelling drug-drug combination responses

We briefly introduce in this section the modelling approach (first proposed using splines by Cremaschi *et al.* 2019), based on a probabilistic interpretation of the cell viability experiment for the combination of two drugs given at specified concentrations.

Each concentration-response experiment can be described by introducing the set of covariates x_{1i}, for $i = 0, \dots, n_1$, and x_{2j}, for $j = 0, \dots, n_2$, indicating the concentration values at which the two compounds are dispensed for a fixed time period. Specifically, $i = 0$ and $j = 0$ correspond to experiments without one or both of the compounds. We point out that the experimental output of a viability experiment is the proportion of viable cells present at the moment of sampling. However, due to measurement error in the experiment and in the control experiment used to normalize the cell counts, these values are often observed outside the range $(0, 1)$, and are therefore modelled as Gaussian random variables

$$Y_{ij}^r \sim N(p_{ij}, \sigma_\epsilon^2), \quad r = 1, \dots, n_{rep}, \tag{2}$$

where n_{rep} is the number of (biological or technical) replicates available for the given combination of drug concentrations indicated by the ordered pair (i, j), $p_{ij} = p_{ij}(x_{1i}, x_{2j})$ is assumed equal over the replicates and indicates the probability of observing a viable cell in the experiment, while σ_ϵ^2 represents the variability of a homoscedastic noise term. The success probabilities p_{ij} are split into a baseline (also called zero interaction) component p_{ij}^0 and an interaction component Δ_{ij}, such that $p_{ij} := p_{ij}^0 + \Delta_{ij}$. The first term is modelled parametrically, based on assumptions similar to the Bliss independence model of Bliss (1939) and the Zero Interaction Potency (ZIP) formulation of Yadav *et al.* (2015), and such that $p_{ij}^0 := h(x_{1i}|\theta_1)h(x_{2j}|\theta_2)$, where $h(x|\theta)$ is defined in (1), and $\theta_h = (m_h, \lambda_h)$ for $h = 1, 2$. On the other hand, the interaction term Δ_{ij} is allowed a flexible modelling formulation. In the R package **BayeSyneRgy**, we have implemented two choices for Δ_{ij}, either using Bayesian splines or Gaussian processes, respectively.

2.1. Spline model

The use of spline terms in order to incorporate the covariates into the model is a flexible tool for modelling non-monotone and complex surfaces (see de Boor 2001, for a review). A tensor spline can be specified for modelling the interaction term Δ_{ij}, obtained as the cross-product of two univariate cubic B-splines, each defined over a set of K_1 and K_2 knots spanning the range of the $\log_{10}$-concentrations $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. In particular, following the approach of Eilers and Marx (2010), we obtain the B-spline basis from differences of truncated power functions. The spline coefficient matrix $\boldsymbol{C}$ is *a priori* distributed as matrix-Normal with mean matrix $\mathbf{0}$ and second order difference covariance matrices Ψ_{K_1} and Ψ_{K_2}, used to penalise the jumps at the knot values of the tensor product spline. Let $B_l(x)$ be a univariate B-spline at knot l evaluated at $x \in \mathbb{R}$:

$$\begin{aligned} B_{ij} &= \gamma_0 + \gamma_1 x_{1i} + \gamma_2 x_{2j} + \mathcal{B}(x_{1i}, x_{2j}), \\ \mathcal{B}(x_{1i}, x_{2j})|\boldsymbol{C} &= \sum_{l,m} \boldsymbol{C}_{lm} B_l(x_{1i}) B_m(x_{2j}), \\ \boldsymbol{C} &\sim \text{Matrix-N}_{K_1,K_2}(\mathbf{0}, \Psi_{K_1}, \Psi_{K_2}). \end{aligned} \tag{3}$$

2.2. Gaussian process model

An alternative flexible modelling approach for the interaction term Δ_{ij} is to use Gaussian processes (GP) (Williams and Rasmussen 2006). Consider an infinite sequence of random variables on $\mathbb{R}^p$ such that $\boldsymbol{X}_i = f(\boldsymbol{x}_i)$, for $i \geq 1$ and for some functional f. The sequence $f(\boldsymbol{x})$ generated for each $\boldsymbol{x} \in \mathbb{R}^p$ is a Gaussian process iff its finite-dimensional laws are Gaussian, that is iff $f(\boldsymbol{x}) \sim \text{N}_p(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}'))$, where $\boldsymbol{\mu}(\boldsymbol{x}) = \mathbb{E}(f(\boldsymbol{x}))$ and $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}') = cov(f(\boldsymbol{x}), f(\boldsymbol{x}'))$. The choice of the kernel function $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}')$ is crucial in the definition of the Gaussian process law. In particular, in the **BayeSyneRgy** package, we have adopted the squared exponential kernel $\boldsymbol{K}^E$ (which is also called radial basis function kernel or

Gaussian kernel), the Matérn kernel $\boldsymbol{K}^M_\nu$, and the rational quadratic kernel $\boldsymbol{K}^R_\alpha$, defined as follows:

$$\begin{aligned}
\boldsymbol{K}^E(\boldsymbol{x},\boldsymbol{y}) &= \exp\left(-\frac{||\boldsymbol{y}-\boldsymbol{x}||^2}{2\ell^2}\right),\\
\boldsymbol{K}^M_\nu(\boldsymbol{x},\boldsymbol{y}) &= \frac{2^{\nu-1}}{\Gamma(\nu)}\left(\sqrt{2\nu}\frac{||\boldsymbol{y}-\boldsymbol{x}||}{\ell}\right)^\nu \mathcal{K}_\nu\left(\sqrt{2\nu}\frac{||\boldsymbol{y}-\boldsymbol{x}||}{\ell}\right),\\
\boldsymbol{K}^R_\alpha(\boldsymbol{x},\boldsymbol{y}) &= \left(1+\frac{||\boldsymbol{y}-\boldsymbol{x}||^2}{2\alpha\ell^2}\right)^{-\alpha},
\end{aligned}$$

where $||\boldsymbol{x}||$ represents the Euclidian norm of the vector $\boldsymbol{x}$. The special function $\mathcal{K}_\nu(z)$ is the modified Bessel function of the second kind of degree ν with positive argument z. The length-scale parameter $\ell > 0$ is of particular importance, since it describes the way in which the distances between data points affect the model. Finally, $\alpha > 0$ is a parameter controlling the relative weighting of large and small distances between the data points in the rational quadratic kernel.

We specify the matrix $\mathcal{B}(x_{1i}, x_{2j})$ by imposing a Gaussian process prior on its vectorised form:

$$\begin{aligned}
&B_{ij} = \gamma_0 + \gamma_1 x_{1i} + \gamma_2 x_{2j} + \mathcal{B}(x_{1i}, x_{2j}),\\
&vec(\mathcal{B}(\boldsymbol{x}_1,\boldsymbol{x}_2))|\boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}'),\sigma_f^2 \sim GP(\boldsymbol{0},\sigma_f^2\boldsymbol{K}(\boldsymbol{x},\boldsymbol{x}')),
\end{aligned} \tag{4}$$

where the covariate space is $\mathbb{R}^2$, and $\boldsymbol{x} = (x_{1i}, x_{2j})$, for $i = 0,\dots,n_1$ and $j = 0,\dots,n_2$. The additional parameter σ_f^2 is a scale parameter reflecting the amount of variability that we would have in the covariate-independent case. It can either be fixed, or one can specify a (weakly informative) hyper-prior distribution to robustify the inference. Both options are available in the **BayeSyneRgy** package, where both, inverse Gamma and half-Cauchy hyper-priors are implemented (Gelman *et al.* 2006).

2.3. Full Bayesian Model for Drug-Drug Interaction

Notice that the domain of Δ_{ij} is the interval $I_{\Delta_{ij}} := (-p^0_{ij}, 1-p^0_{ij})$, since the term p_{ij} represents a probability and as such must lie in $[0,1]$. Hence, a suitable transformation $g:\mathbb{R}\to I_{\Delta_{ij}}$ of the building block B_{ij} is needed, in order to maintain these constraints. Several choices for g can be considered, and here we use the following:

$$g(B_{ij}) = -p^0_{ij}(1+e^{b_1 B_{ij}})^{-1} + (1-p^0_{ij})(1+e^{-b_2 B_{ij}})^{-1}.$$

The link function g is applied to each spline term and pair of concentrations tested, yielding $\Delta_{ij} = g(B_{ij})\mathbb{I}_0(i,j)$, where $\mathbb{I}_0(i,j) = 0$ if $i = 0$ or $j = 0$, and $\mathbb{I}_0(i,j) = 1$ otherwise. This indicator prevents any interaction in the absence of either of the compounds, i.e. for x_{10} or x_{20}.

In summary, the full model is specified as follows:

$$\begin{aligned}
&Y^r_{ij} \overset{\text{ind}}{\sim} N(p_{ij},\sigma^2_\epsilon), \quad p_{ij} = p^0_{ij} + \Delta_{ij}\\
&p^0_{ij} = h(x_{1i}|m_1,\lambda_1)h(x_{2j}|m_2,\lambda) = \left(1+\left(\frac{x_{1i}}{m_1}\right)^{\lambda_1}\right)^{-1}\left(1+\left(\frac{x_{2j}}{m_2}\right)^{\lambda_2}\right)^{-1},\\
&\Delta_{ij} = g(B_{ij})\mathbb{I}_0(i,j), \quad g(B_{ij}) = -p^0_{ij}\left(1+e^{b_1 B_{ij}}\right)^{-1} + \left(1-p^0_{ij}\right)\left(1+e^{-b_2 B_{ij}}\right)^{-1},\\
&B_{ij} = \gamma_0 + \gamma_1 x_{1i} + \gamma_2 x_{2j} + \mathcal{B}\left(x_{1i}, x_{2j}\right),\\
&\psi|\alpha_\psi,\beta_\psi \sim \text{Gamma}(1,1), \quad \psi \in \{\lambda_1,\lambda_2,b_1,b_2\},\\
&\phi|\sigma^2_\phi \sim \text{N}(0,\sigma^2_\phi), \quad \phi \in \{m_1,m_2,\gamma_0,\gamma_1,\gamma_2\},\\
&\sigma^2_{m_1},\sigma^2_{m_2},\sigma^2_{\gamma_0},\sigma^2_{\gamma_1},\sigma^2_{\gamma_2},\sigma^2_\epsilon \sim \text{Inverse-Gamma}(3,2),
\end{aligned} \tag{5}$$

for $i = 0,\dots,n_1$, $j = 0,\dots,n_2$, and $r = 1,\dots,n_{rep}$. The transformed interaction term $\mathcal{B}(x_{1i}, x_{2j})$ is distributed according to (3) or (4). The hyperpriors for the positive and variance parameters are chose according to standard use, assuming prior with unitary mean and variance, and that are (when possible) conjugate. As it will be shown later, other choices of hyperprior distributions are also available in the R package. In particular, for all scale parameters $(\sigma^2_{m_1}, \sigma^2_{m_2}, \sigma^2_{\gamma_0}, \sigma^2_{\gamma_1}, \sigma^2_{\gamma_2}, \sigma^2_{\epsilon})$ half-Cauchy priors are provided in addition to the conjugate inverse Gamma priors.

3. Demo example

The BayesSyneRgy package can be installed from GitHub (https://github.com/ltronneb/bayesynergy), for example as follows:

```
R> library(devtools)
R> install_github("ltronneb/bayesynergy")
```

We proceed by presenting a quick demonstration of the use of the **BayeSyneRgy** package for the study of drug-drug combinations. We illustrate the use of the package on data from Mathews Griner *et al.* (2014), which was also used in the vignette of the R package **synergyfinder** (He *et al.* 2018). In the original combination study, approximately 500 drugs were combined with the Bruton's kinase inhibitor ibrutinib, an anti-cancer drug that has proven promising in previous studies (Vela, McBride, Jaglowski, and Andritsos 2016). Each drug combination was tested in-vitro on a diffuse large B-cell lymphoma (DLBCL) cell line called TMD8. In-vitro experiments were performed over a 6x6 matrix of concentrations, with ibrutinib ranging in $\boldsymbol{x}_2 = (0, 0.2, 0.78, 3.12, 12.5, 50)$ nM. The compounds selected for this example are canertinib and ispinesib (for both, $\boldsymbol{x}_1 = (0.00, 9.77, 39.06, 156.25, 625, 2500)$ nM), depicting two very different response behaviours. All experiments were performed without replicates ($n_{rep} = 1$).

First, we load the **BayeSyneRgy** package and the data, which was imported from the **synergyfinder** package and reformatted to match the requirements of our package.

```
R> library("BayeSyneRgy")
R> data("mathews_DLBCL")
```

The two combination experiments are organized into a nested list of length two, each list element containing the data for one of the two experiments (ibrutinib/ispinesib and ibrutinib/canertinib). The data for each experiment are stored as a list of two matrices, one containing the normalized response values and one containing the corresponding drug concentrations. The following shows some of the data points for the ispinesib/ibrutinib combination experiment.

```
R> head(mathews_DLBCL[[1]][[1]])

     Viability
[1,] 1.2295618
[2,] 1.0376006
[3,] 1.1813851
[4,] 0.5882688
[5,] 0.4666700
[6,] 0.2869514

R> head(mathews_DLBCL[[1]][[2]])

     ibrutinib ispinesib
[1,]    0.0000         0
[2,]    0.1954         0
[3,]    0.7812         0
[4,]    3.1250         0
[5,]   12.5000         0
[6,]   50.0000         0
```

We fit our model using the function `BayeSyneRgy()`, taking as input the response data and the concentration pairs extracted from the data list. Default settings are applied, which means that the model is fitted using a Gaussian process with a squared exponential kernel to model the interaction term in a flexible manner. The kernel length-scale parameter ℓ is assumed random and $\ell \sim \Gamma(1,1)$, while the parameter σ_f^2 is fixed to 1. Default inverse-Gamma$(3,2)$ hyper-priors with mean and variance 1 are used for all the variance parameters.

```
R> #Set random seed for reproducibility
R> set.seed(1234)
R> #Fit ispinesib + ibrutinib
R> y_mat1 <- mathews_DLBCL[[1]][[1]]
R> x_mat1 <- mathews_DLBCL[[1]][[2]]
R> GP_Out_Combo1 <- BayeSyneRgy(y_mat1, x_mat1)

Ec50_1 acc. rate = 0.249501
Ec50_2 acc. rate = 0.300988
Slope_1 acc. rate = 0.28868
Slope_2 acc. rate = 0.335042
gamma_0 acc. rate = 0.255765
gamma_1 acc. rate = 0.263336
gamma_2 acc. rate = 0.239065
b acc. rate = 0.116691
B acc. rate = 0.198
ell acc. rate = 0.233172

R> #Fit canertinib + ibrutinib
R> y_mat2 <- mathews_DLBCL[[2]][[1]]
R> x_mat2 <- mathews_DLBCL[[2]][[2]]
R> GP_Out_Combo2 <- BayeSyneRgy(y_mat2, x_mat2)

Ec50_1 acc. rate = 0.262991
Ec50_2 acc. rate = 0.264509
Slope_1 acc. rate = 0.305339
Slope_2 acc. rate = 0.272014
gamma_0 acc. rate = 0.268101
gamma_1 acc. rate = 0.28175
gamma_2 acc. rate = 0.251969
b acc. rate = 0.09125
B acc. rate = 0.198
ell acc. rate = 0.233678
```

The output of the function `BayeSyneRgy()` is an object of `S3` class `BayeSyneRgy`. At the end of each MCMC run, the posterior acceptance rates are displayed on screen, only for those parameters that are non-conjugate in the proposed model. These values can be useful to the user for a first check of the performance of the MCMC run. Very low acceptance rates can be symptomatic of not-yet-reached convergence to the target distribution of the Markov chain, or of difficulty of the model to fit the data at hand, e.g., if the data set presents a relevant proportion of responses outside the range $(0,1)$.

The posterior samples can be summarized with the following `summary()` function, which produces a table containing some posterior descriptive quantities for the measures of interest, which are of importance in order to understand the effectiveness of the drugs in the experiment. In particular, we report the Drug Sensitivity Score (DSS) (Yadav *et al.* 2014) as a measure of efficiency for the individual drugs as well as measures for the drug combination, such as the relative Volume Under the Surface (rVUS) for the posterior distribution of the estimated concentration response surface $\hat{p}_{ij}$, for the interaction part $\hat{\Delta}_{ij}$ and for the positive and negative interaction parts (representing antagonistic and synergistic effects, respectively).

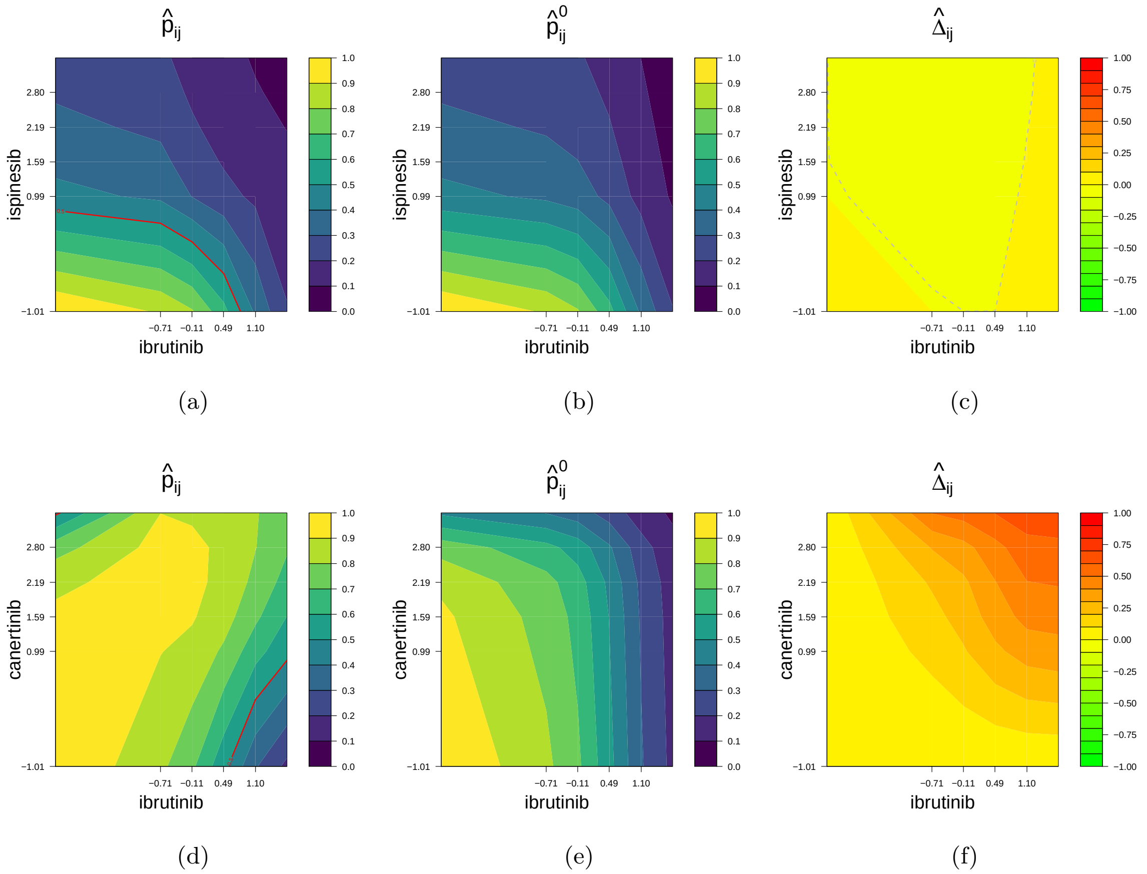
```
R> summary(GP_Out_Combo1)

                     mean    sd   2.5 %  50 % 97.5 % ESS/N
EC50 (ibrutinib)     0.80  0.57 -0.3509  0.81   1.88  0.88
EC50 (ispinesib)     0.21  0.93 -1.6313  0.22   1.98  0.96
Slope (ibrutinib)    1.05  0.85  0.0769  0.86   3.24  0.84
Slope (ispinesib)    0.27  0.44  0.0067  0.14   1.55  1.00
DSS (ibrutinib)     25.16 10.61  7.2090 24.07  49.38  0.81
DSS (ispinesib)     57.59 10.20 41.1927 55.45  84.76  0.93
S2_EPS               0.12  0.03  0.0770  0.12   0.19  1.00
rVUS_p              62.75  5.62 52.4311 62.61  74.21  0.93
rVUS_Delta           7.78  3.95  2.1000  7.17  17.55  1.00
rVUS_syn             4.31  4.25  0.0000  3.18  14.45  0.84
rVUS_ant             3.57  4.40  0.0000  1.89  15.80  0.88

log-Pseudo Marginal Likelihood (LPML) =  0.7386405

R> summary(GP_Out_Combo2)

                      mean    sd   2.5 %  50 % 97.5 % ESS/N
EC50 (ibrutinib)      0.47  0.76 -1.1529  0.54   1.77  0.50
EC50 (canertinib)     3.28  1.34 -0.1615  3.30   5.76  0.34
Slope (ibrutinib)     1.11  1.01  0.0378  0.85   3.76  0.80
Slope (canertinib)    1.58  1.19  0.0271  1.35   4.64  0.44
DSS (ibrutinib)      31.34 13.74  7.5807 30.23  58.12  0.38
DSS (canertinib)     11.96 15.22  0.0019  6.63  54.94  0.18
S2_EPS                0.33  0.09  0.1990  0.31   0.55  0.44
rVUS_p               19.14  6.64  8.5214 18.42  34.44  0.35
rVUS_Delta           20.36 10.32  5.9671 18.39  44.19  0.26
rVUS_syn              0.67  1.40  0.0000  0.00   4.99  0.81
rVUS_ant             21.53 11.66  4.9127 19.30  46.92  0.25

log-Pseudo Marginal Likelihood (LPML) =  -28.41531
```

Note that the combination of ibrutinib and ispinesib has a better goodness-of-fit than the ibrutinib/canertinib one, as measured by the log-Pseudo Marginal Likelihood (LPML) value which is printed in the `summary()` output after the summary table. The LPML value, together with the posterior estimates of σ_ϵ^2 (`S2_EPS`), can provide information about the quality of the data set. In particular, these measures can reflect situations in which the data are very noisy, or where the model does not fit the data well, because of problems due to normalization using control experiments, which can result in technical bias and in an unusally large number of (normalized) response values recorded outside the range $(0, 1)$.

The posterior quantiles reported in `summary()` can be interpreted as a measure of certainty about the presence of synergy or antagonism. In particular, the 95% equal-tailed posterior credible interval for `rVUS_syn` contains zero in both experiments, which indicates that we remain uncertain about the presence of synergy in this data set. The corresponding interval for `rVUS_ant` however, does not include zero for the ibrutinib/canertinib experiment, which provides evidence for the presence of an antagonistic effect in this experiment.

```
R> plot(GP_Out_Combo1)
R> plot(GP_Out_Combo2)
```

In order to better visualize the results from fitting the model, the posterior distributions of individual parameters summarizing both monotherapy (Figure 2, top row) and combination therapy (Figure 3)



(a) (b) (c)

(d) (e) (f)

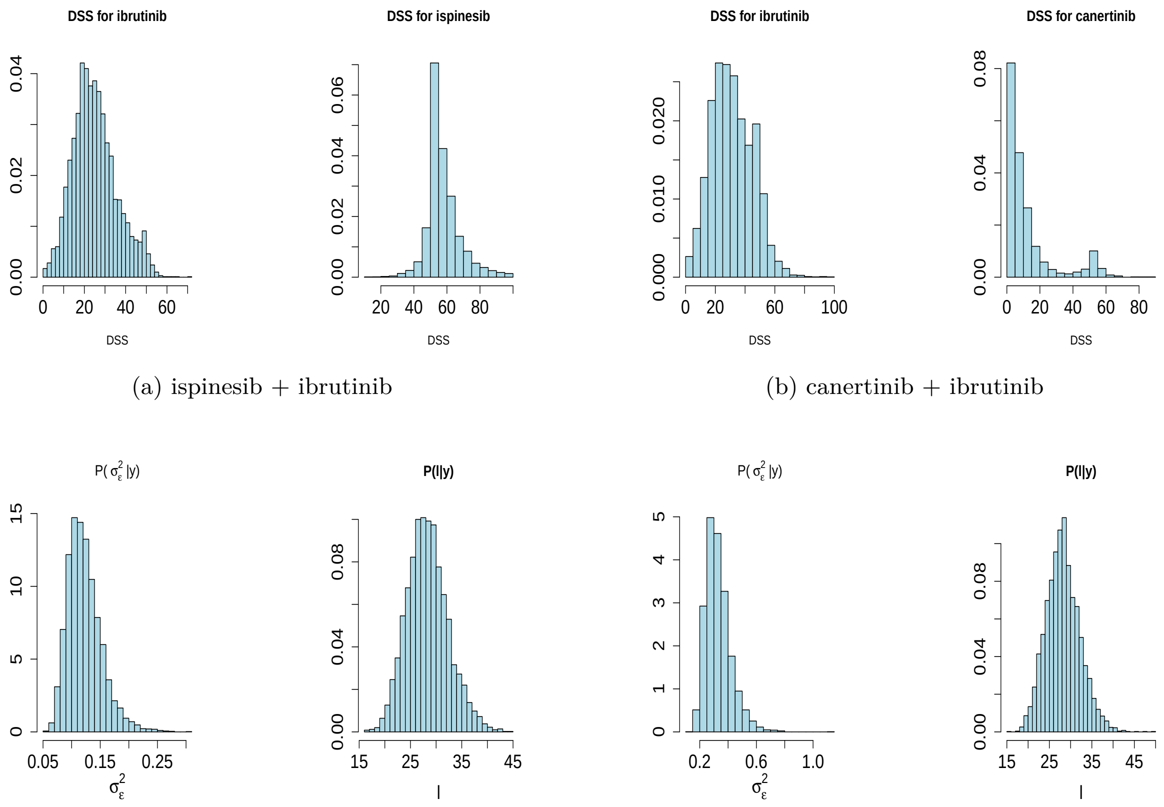
Figure 1: DLBCL data – Contour plots of the posterior mean of the surfaces of interest for combinations of ispinesib with ibrutinib (top row) and canertinib with ibrutinib (bottom row). The red lines in the left-most plots indicate the contour lines at 0.5, i.e. all concentration combinations where the posterior mean of p_{ij} is equal to 0.5, which represents a 50% reduction in viability is predicted. The grey dashed line indicates the level curve corresponding to null interaction.

effectiveness and some hyper-parameters (Figure 2, bottom row) as well as posterior mean surfaces for the total response p_{ij}, zero interaction p^0_{ij} and interaction Δ_{ij} (Figure 1) are plotted via the `BayeSyneRgy` class method `plot()`.

We first consider each drug alone and look at the posterior distributions of their DSS values in Figure 2. As expected, the behaviour of ibrutinib alone is consistent in the two experiments, and agrees with findings in the literature that present it as an effective anti-cancer drug (median DSS = 24.07 and 30.23, respectively). However, we do observe a much higher DSS score for the drug ispinesib alone (median DSS = 55.45), which indicates a very good efficiency of ispenisib as a mono-therapy.

We also report in Figure 3 the posterior distributions of the relative Volume Under the Surface (rVUS) for several surfaces. This measure conveys different information depending on the surface being considered. For instance, when choosing the surface $(1-p_{ij})$, we obtain a measure of the overall efficacy of the drug combination. Similarly, we can consider the surface $|\Delta_{ij}|$, and thereby compute a measure of the total interaction.

Alternatively, we can distinguish between its synergistic (Δ^-_{ij}) and antagonistic (Δ^+_{ij}) components, which are simply the negative and positive parts of the interaction surface Δ_{ij}, respectively. We can see that the combination of ibrutinib and ispenisib is very effective, and retains some synergistic interaction. On the other hand, the combination of ibrutinib and canertinib is less effective, and in addition it presents most of its interaction as antagonistic. The full posterior distributions as summarized in Figure 3 provide further insight into the uncertainty around the estimates for synergistic or antagonistic effects and supplement our initial observations based on the 95% equal-tailed posterior credible intervals obtained with `summary()`.



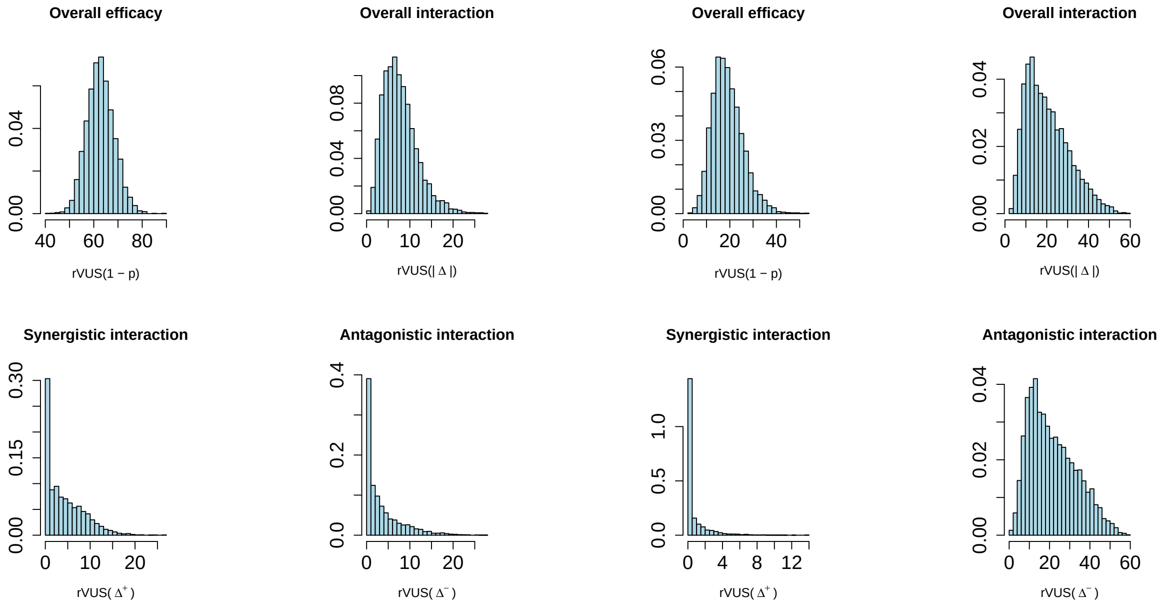
(a) ispinesib + ibrutinib

(b) canertinib + ibrutinib

(c) ispinesib + ibrutinib

(d) canertinib + ibrutinib

Figure 2: DLBCL data – Histograms of the posterior distributions of the drug sensitivity scores (DSS) for each drug (top row) in both combination experiments and of the corresponding posterior distributions of the homoscedastic variance σ_ϵ^2 and kernel length-scale ℓ (bottom row).



(a) ispinesib + ibrutinib

(b) canertinib + ibrutinib

Figure 3: DLBCL data – Histograms of the posterior distributions of the relative Volume Under the Surface (rVUS) values for different surfaces.

This data set was also analyzed in the vignette of the package **synergyfinder** using the ZIP model, where an overall synergistic effect was reported for the ibrutinib/ispinesib experiment and an overall antagonistic effect was reported for the ibrutinib/canertinib combination. Our results allow a more nuanced conclusion, since our estimates come with measures of uncertainty, which tell us that the seeming synergistic effect for ibrutinib/ispinesib is quite weak and cannot be distinguished from zero.

Note that the full posterior chains of all these quantities are also returned to the user in the object of `S3 BayeSyneRgy`. Therefore, it is easily possible to do further analyses based on the MCMC results or to perform more detailed MCMC diagnostics, e.g. show traceplots of the MCMC chains for individual parameters to check their convergence to the target distributions. For instance, via the popular **coda** package, one can plot the traceplot of σ^2_ϵ as follows (see Figure 4).

```
R> library(coda)
R> plot(as.mcmc(GP_Out_Combo1$OUTPUT$MCMC_Output$S2_EPS))
```

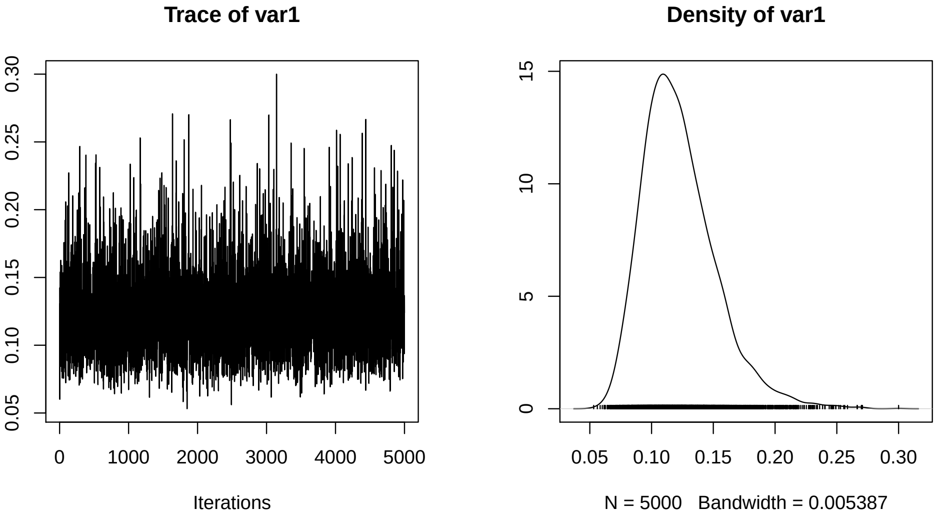


Figure 4: DLBCL data – Example MCMC diagnostics output after easy transformation of the MCMC output to an `mcmc` object from the **coda** package: Traceplot and histogram of σ^2_ϵ for the ispinesib/ibrutinib combination experiment.

4. Computational details

Posterior estimates of the parameters of interest are obtained by Markov Chain Monte Carlo (MCMC) sampling. The non-conjugate parameters appearing in the model are updated via Metropolis-within-Gibbs steps, of which we use an adaptive version (see Griffin and Stephens 2013, algorithms 5 and 6). The default settings, that can be changed by providing the new values within a list, are the following:

```
R> Alg_param.default <- list(n_burn = 50000, thin = 10, n_save = 5000, g0 = 1000,
+                                   wg = 0.7, opt_rate = 0.234, eps = 0.001)
```

The first three elements of the list contain the number of iterations to be performed as burn-in period (including the adaptive part), the thinning, and the number of saved iterations. The other parameters are used for the adaptation part of the Metropolis-Hastings steps, and are the adaptive burn-in (`g0`), the exponent of the adaptive function (`wg`), the optimal acceptance rate (`opt_rate`), and a parameter included for numerical stability (`eps`).

The default parameters guiding the MCMC sampler are chosen carefully and should work well for typical datasets. Nevertheless, we advice experienced users to check the MCMC quality measures, e.g.

posterior acceptance rates and relative effective sample sizes, which we provide for evaluation of the properties of the MCMC chains. For more detailed MCMC diagnostic checks, it is straightforward to convert the MCMC output into `coda::mcmc` objects, which then allows the use of the entire functionality of the popular R package **coda** for summarizing and plotting the MCMC output, as well as diagnostic tests of convergence to the equilibrium distribution of the Markov chain.

The implementation of the MCMC sampler is done in C++ for computational speed; the integration between R and C++ code is achieved with the **Rcpp** (Eddelbuettel and François 2011), **RcppArmadillo** (Eddelbuettel and Sanderson 2014) and **BH** (Eddelbuettel, Emerson, and Kane 2019) R packages.

5. The BayeSyneRgy main functions and arguments

The function `BayeSyneRgy()` is the main function of the **BayeSyneRgy** package and provides the interface for communication with the MCMC sampler. The function requires as arguments a matrix `y_mat` of responses of the biological sample to the drugs in the drug combination experiment, a corresponding matrix `x_mat` of unique concentrations of each drug in the drug combination experiment, and a specification `type` of which model to use. For each model type, default parameters are set inside the function, but these can be overwritten by passing parameters in corresponding lists. Table 1 shows the arguments that can be passed to the `BayeSyneRgy()` function, with default values indicated and set in bold.

Table 1: Arguments of the `BayeSyneRgy()` function. Default values are indicated and set in bold.

Argument	Description	Values
`y_mat`	The matrix of responses	-
`x_mat`	The matrix of corresponding pairs of drug concentrations	-
`log10_conc`	logical; if true, concentrations in `x_mat` are assumed given on $\log_{10}$ scale.	`TRUE` / `FALSE` (default)
`type`	The model specification	1 (Splines), **2 (GP with squared exponential kernel)** (default), 3 (GP with Matérn kernel), 4 (GP with rational quadratic kernel)
`Alg_param`	List of parameters for the MCMC sampler	See Table 2
`Hyper_param`	List of hyperparameters for the prior distributions	See Table 2
`GP_param`	List of hyperparameters for the Gaussian process' kernel.	See Table 2
`var_prior`	A specification of what prior to put on variance hyperparameters	**1 (Inverse-Gamma)** (default), or 2 (Half-Cauchy)

[1]

Inside the `BayeSyneRgy()` function, default values are set for all hyperparameters of the model. Table 2 shows valid variable names of these lists. Any parameter passed by the user will overwrite the default values. Note that for the `Hyper_param` list, each parameter must have a prefix indicating which hyperparameters to change. The Inverse-Gamma (IG) and Gamma (G) priors both use the prefixes `a_` and `b_`, while the Half-Cauchy uses `h_`. For example, to change the hyperparameters for the Gamma prior on `Slope_1` to specify $\lambda_1 \sim \text{Gamma}(4,4)$, and at the same time to change the Half-Cauchy scale parameter to specify $\sigma^2_\epsilon \sim \text{Half-Cauchy}(2)$, simply run the model as follows:

```
R> hyp_par <- list(a_Slope_1=4, b_Slope_1=4, h_s2_eps=2)
R> fit <- BayeSyneRgy(y_mat, x_mat, Hyper_param=hyp_par, var_prior=2)
```

Note the `var_prior=2` option, which denotes that a Half-Cauchy prior should be used for the σ^2_ϵ parameter.

We also allow for prior distributions on the Gaussian process kernel parameters, shown in the `GP_param` list in Table 2. For `ell`, `nu` and `alpha`, the choice to put a prior on these will be inferred from the value passed. If a scalar value is passed, the parameters are assumed fixed at the passed value, while a vector of length two will specify a prior as indicated in the table. For the kernel amplitude `sigma2_f`, a vector of length two will be interpreted as the hyperparameters of an Inverse-Gamma prior distribution, while a single scalar will need the extra indication of `sigma2_f_prior` to indicate whether the user wants a Half-Cauchy prior with one hyper-prior parameter, or a fixed parameter value. As an example, the following code will put a Gamma(1, 1) prior on the α parameter of the rational quadratic kernel, while also keeping the length scale, ℓ, fixed to 0.5.

```
R> gp_par <- list(ell=0.5, alpha=c(1,1))
R> fit <- BayeSyneRgy(y_mat, x_mat, GP_param=gp_par)
```

Table 2: List of optional parameters that can be passed to the `BayeSyneRgy()` function.

List	**Parameter**	**Description**	**Default**	**Priors** (default in bold)
`Alg_param`	`n_burn`	number of burn-in iterations for the MCMC sampler	50 000	-
	`thin`	amount of thinning	10	-
	`n_save`	number of final iterations returned to the user	5000	-
	`g0`	adaptive burn-in	1000	-
	`wg`	exponent of the adaptive function	0.7	-
	`opt_rate`	optimal acceptance rate	0.234	-
	`eps`	parameter used for numerical stability	0.001	-
`Hyper_param`	`a_Slope_1`, `b_Slope_1`	Hyper-parameters for monotherapy slope of drug 1	$(a, b) = (1, 1)$	**Gamma(a,b)**
	`a_Slope_2`, `b_Slope_2`	Hyper-parameters for monotherapy slope of drug 2	$(a, b) = (1, 1)$	**Gamma(a,b)**
	`a_b1`, `b_b1`	Hyper-parameters for monotherapy lower asymptote of drug 1	$(a, b) = (1, 1)$	**Gamma(a,b)**
	`a_b2`, `b_b2`	Hyper-parameters for monotherapy lower asymptote of drug 2	$(a, b) = (1, 1)$	**Gamma(a,b)**
	`a_s2_Ec50_1`, `b_s2_Ec50_1`, `h_s2_Ec50_1`	Hyperparameters for the variance of the EC-50 for drug 1	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
	`a_s2_Ec50_2`, `b_s2_Ec50_2`, `h_s2_Ec50_2`	Hyperparameters for the variance of the EC-50 for drug 2	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
	`a_s2_gamma_0`, `b_s2_gamma_0`, `h_s2_gamma_0`	Hyperparameters for the variance of the gamma_0 parameter	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
	`a_s2_gamma_1`, `b_s2_gamma_1`, `h_s2_gamma_1`	Hyperparameters for the variance of the gamma_1 parameter	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
	`a_s2_gamma_2`, `b_s2_gamma_2`, `h_s2_gamma_2`	Hyperparameters for the variance of the gamma_2 parameter	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
	`a_s2_eps`, `b_s2_eps`, `h_s2_eps`	Hyperparameters for the noise term	$(a, b) = (3, 2)$, $h = 1$	**Inv-Gamma(a,b)**, Half-Cauchy(h)
`GP_param`	`ell`	The length scale used in all kernels.	$(a, b) = (1, 1)$	fixed, **Gamma(a,b)**
	`nu`	The nu parameter for the Matérn kernel.	5/2	**fixed**, Gamma(a,b)
	`alpha`	The alpha parameter for the rational quadratic kernel.	1	**fixed**, Gamma(a,b)
	`sigma2_f`	The kernel amplitude, i.e. $\sigma_f^2 k(\boldsymbol{x}, \boldsymbol{x}')$	1	**fixed**, Inv-Gamma(a,b), Half-Cauchy(h)
	`sigma2_f_prior`	An indicator for the type of prior on `sigma2_f`. Either 0 (fixed), 1 (Inv-Gamma), or 2 (Half-Cauchy)	0	-

The output of the `BayeSyneRgy()` function will be an object of S3 class `BayeSyneRgy`. We provide `S3` methods `plot` and `summary` for the class `BayeSyneRgy`. The class-unique arguments that are accepted by these functions are listed in Table 3.

The `plot()` function is extended to plot response and interaction surfaces, either as two-dimensional contour surfaces, or in a three-dimensional interactive form (argument `plot_type = "2D"`, `"3D"`, or `"both"`). In the 2D version, specific contour lines can be added, for example the 0.5 contour line in the total response surface, which can be interpreted as an alternative for drug combination experiments to the well-established EC_{50} measure for mono-therapy experiments. In addition, `plot()` provides summaries in form of histograms of the DSS for the monotherapies, rVUS and similar measures as detailed in section 3. All plots can be saved as PDF graphics (argument `save_plot=TRUE`) or shown interactively.

The `summary()` function prints relevant summary statistics for the marginal posterior distributions for measures of interest. This includes the monotherapy measure Drug Sensitivity Score (DSS) (Yadav *et al.* 2014), as well as measures inherent to the drug combination, such as the relative Volume Under the Surface (rVUS) for the posterior distribution of the estimated concentration response surface $\hat{p}_{ij}$, for the interaction part $\hat{\Delta}_{ij}$ and for the positive and negative interaction parts (representing antagonistic and synergistic effects). For more details on these measures see Cremaschi *et al.* (2019). The posterior distributions are summarized by their mean, standard deviation and 2.5%, 50%, and 97.5% quantiles, respectively. In the last column, the Effective Sample Size is listed, i.e. the MCMC sample size adjusted for autocorrelation, over the number of MCMC iterations saved from the Gibbs sampler (ESS/N). This quantity measures the goodness of the mixing of the Markov chain of the posterior samples for the respective parameter, where values close to one represent good mixing, while values close to zero are indicative of poor mixing. In addition, the log-Pseudo Marginal Likelihood (LPML) is listed to provide a measure of goodness of fit.

Table 3: Arguments of the `plot()` and `summary()` methods for the `BayeSyneRgy` class

Function	**Argument**	**Description**
`plot.BayeSyneRgy()`	`x`	An object of class `BayeSyneRgy`.
	`add_contour`	logical; if TRUE draw contour lines at the specified level in the 2D response surface. Default TRUE.
	`contour_levels`	numeric; vector of levels for drawing contours. Default 0.5.
	`plot_type`	character; denotes how to display the surface plots. Either "2D", "3D", or "both". Default "2D".
	`save_plot`	logical; if TRUE save plots to a PDF file. Default FALSE.
`summary.BayeSyneRgy()`	`object`	An object of class `BayeSyneRgy`.

6. Simulation study

In this section, we investigate the proposed models further on simulated data. The models are compared in terms of goodness-of-fit, as well as computational performance. Synthetic viability data are simulated according to the following data-generating model, for $r = 1, \dots, n_{rep} = 3$:

$$
\begin{aligned}
y_{ij}^r | p_{ij}, \sigma_\epsilon^2 \overset{\text{ind}}{\sim} \mathrm{N}(p_{ij}, \sigma_\epsilon^2), \quad \sigma_\epsilon^2 = 0.005, \quad p_{ij} = p_{ij}^0 + \Delta_{ij}, \\
p_{ij}^0 = \mathbf{\Phi}\left(2\boldsymbol{\mu} - (x_{1i}, x_{2j}), \boldsymbol{\mu} = (0,5), \Sigma = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}\right), \\
\Delta_{ij} = 0.5\left(\mathbf{\Phi}\left((x_{1i}, x_{2j}), \boldsymbol{\mu}_1 = (1,1), \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\right. \\
\left. - \mathbf{\Phi}\left(-2\boldsymbol{\mu}_1 - (x_{1i}, x_{2j}), \boldsymbol{\mu}_2 = -\boldsymbol{\mu}_1, \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\right),
\end{aligned}
$$

with concentrations $\log_{10}(\boldsymbol{x}_1) = (-\infty, -3, -2, \dots, 4)$ and $\log_{10}(\boldsymbol{x}_2) = (-\infty, -4, -2.5, \dots, 3.5)$. The first concentration represents the absence of the drug in $\log_{10}$-scale. The choice of the baseline and interaction terms is made to represent a realistic concentration-response surface for a viability experiment where two drugs are combined. In particular, the interaction term Δ assumes positive values for higher concentrations (antagonism), while some synergistic behaviour is visible at medium-range concentrations.

The simulated data are fitted under model specification (2.2), using a Matérn kernel with parameter $\nu = 5/2$ (default), fixed value of $\ell = 1$, and inv-Gamma$(3,2)$ for σ_f^2 (specified with argument list `GP_param`). These settings can be easily changed by specifying the values of the hyperparameters of ℓ, σ_f^2 or ν (if additional priors are desired). In particular, while a gamma prior distribution can be imposed on the parameter ν, we suggest to keep it fixed in order to avoid numerical issues in the Bessel function computations.

```
R> set.seed(1234)
R> GP_param <- list(ell = 1, sigma2_f = c(3,2))
R> GP_Out <- BayeSyneRgy(y_mat, x_mat, log10_conc = TRUE, type = 3, GP_param = GP_param)
```

```
Ec50_1 acc. rate = 0.236549
Ec50_2 acc. rate = 0.268805
Slope_1 acc. rate = 0.234574
Slope_2 acc. rate = 0.242031
gamma_0 acc. rate = 0.249041
gamma_1 acc. rate = 0.240119
gamma_2 acc. rate = 0.238122
b acc. rate = 0.211607
B acc. rate = 0.230849
```

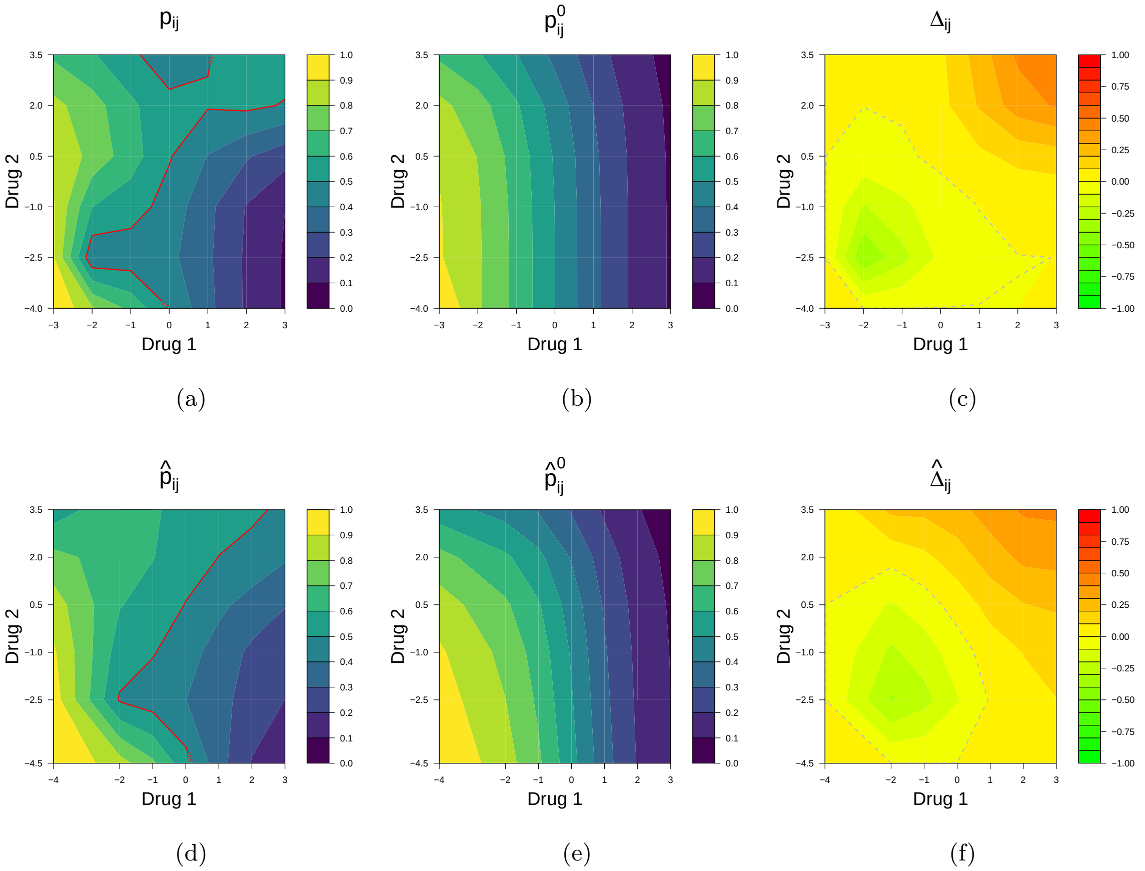
```
R> summary(GP_Out)
```

```
                  mean     sd  2.5 %  50 % 97.5 % ESS/N
EC50 (Drug 1)    0.171 0.3146 -0.457  0.17  0.789  0.89
EC50 (Drug 2)    3.947 0.9416  2.557  3.79  6.246  0.60
Slope (Drug 1)   0.360 0.1190  0.183  0.34  0.634  0.78
Slope (Drug 2)   0.280 0.2542  0.098  0.23  0.809  0.85
DSS (Drug 1)    41.779 3.8621 34.462 41.72 49.561  0.88
DSS (Drug 2)    14.749 6.0508  3.049 14.68 26.979  0.87
S2_EPS           0.041 0.0055  0.031  0.04  0.053  0.72
rVUS_p          47.180 1.8201 43.593 47.17 50.716  0.60
rVUS_Delta      13.058 2.5011  8.629 12.90 18.366  0.67
rVUS_syn         5.085 2.3515  1.152  4.83 10.267  0.80
rVUS_ant         8.415 2.3106  4.290  8.29 13.358  0.86

log-Pseudo Marginal Likelihood (LPML) =  68.87038
```

The results for the chosen model specification are reported in Figure 5, comparing the contour plots of the estimated surfaces with the ones used to simulate the data. We obtain a good fit (LPML = 68.87) and good recovery of the surfaces of interest. In particular, the main regions of interaction are well recovered (see Figure 5(c) and 5(f)). To study the efficacy of the compounds, we also report the posterior distributions of the DSS for the monotherapy drugs in Figure 6(a), and the posterior distributions of the relative volume under the surfaces in Figure 7. It is clear that the two simulated drugs are behaving very differently, with "Drug 1" being more effective than the other. To show all the output of the `plot` function of this package, we report the posterior distributions of σ^2_ϵ, ℓ and $\sigma 2_f$ in Figure 6(b).

A concluding comparison is made with the results obtained using the package **synergyfinder**, where the three replicate y^r_{ij} observations are averaged to a single number at each concentration. Figure 8 shows the interaction surface estimated using the Bliss method in the **synergyfinder**. Contrarily to our setting, here positive values indicate synergy, while negative indicate antagonism, and are scaled in the range (-100,100). The interaction surface is recovered, as well as the range of interaction values.

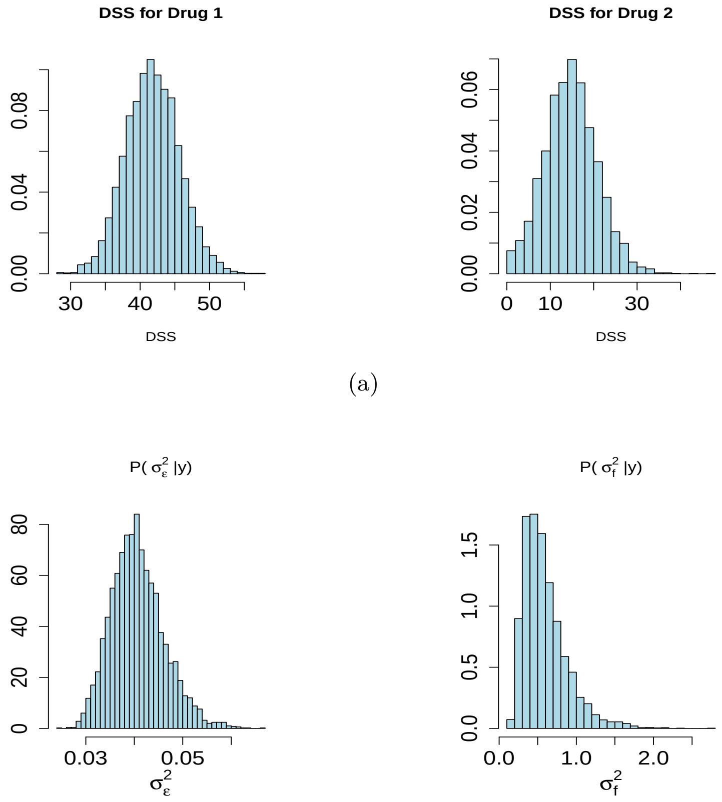
The output of the **synergyfinder** package provides, in addition to the contour and surface plots for Δ, an average of the surface values. This is used as overall measure of the interaction estimated for the combined experiment, as opposed to the hereby proposed rVUS($|\Delta|$) (see Figure 7). The true value of this average for the interaction surface used in the simulations is 0.0522, while the output of **synergyfinder** yields -0.046, after appropriate re-scaling. On the other hand, our method estimates the average Δ value to be 0.0961, with a 95% credibility interval equal to (-0.001,0.19), which includes the truth.



(a) (b) (c)

(d) (e) (f)

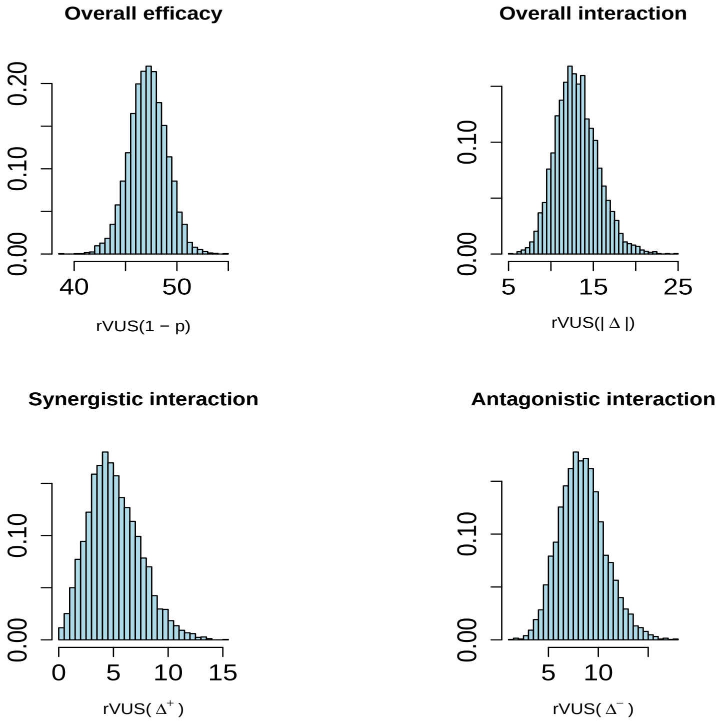
Figure 5: Simulated data – Posterior mean of the surfaces of interest.



(a)

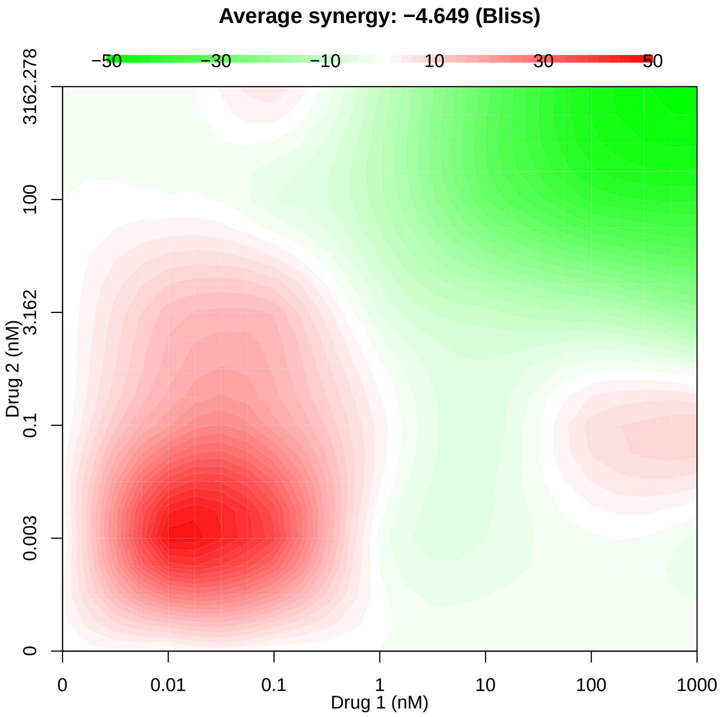
(b)

Figure 6: Simulated data – Top row: Posterior distributions of the drug sensitivity scores (DSS) for each monotherapy drug. Bottom row: Posterior distributions of the homoschedastic variance σ_ϵ^2 and of the parameters of the GP σ_f^2 and ℓ.



(a)

Figure 7: Simulated data – Posterior distributions of the relative Volumes Under the Surface (rVUS) for $(1 - p_{ij})$, $|\Delta|$, Δ^+, Δ^-.



(a)

Figure 8: Simulated data – Interaction surface estimated using the Bliss method via the package **synergyfinder**. This contour plot is the direct output of the `PlotSynergy()` function in that package. Notice how the values of interaction are positive for synergy, and negative for antagonism.

7. Conclusions

The use of multiple anti-cancer drugs in combination treatments has become the standard approach in cancer therapies. A successful combination will yield a better outcome when compared to the effect of the same drugs acting separately. In order to better understand which drug combinations are expected to show such synergistic effects, the efforts in the study of viability experiments for drug combinations are increasing, and the development of mathematical and statistical models to analyse such experimental data is an active research area.

In this work, we presented a new R package **BayeSyneRgy**, which implements a statistical model-based approach to the study of interaction between two drugs, where the drug combination surface is modelled using flexible Bayesian models, based on either splines or Gaussian processes (Cremaschi *et al.* 2019). This approach allows us to address several of the drawbacks of classical synergy models, because our statistical approach takes into account all variability in the data and allows the use of replicate measurements.

In our experience, the splines and Gaussian process models typically result in very similar estimates, if the hyper-priors are chosen in a weakly informative manner. If there are notable differences, this might indicate potential problems with the prior specification, for example that the Gaussian process length scale parameter ℓ is chosen too large, which would then result in overly smooth estimated surfaces.

The **BayeSyneRgy** software performs full posterior inference on all model parameters using an adaptive MCMC sampler, which was implemented in C++ for computatinal efficiency. This makes it possible to investigate the uncertainty associated with all parameters of interest, including measures of efficacy of individual drugs and of the drug combination, as well as measures of overall synergistic and antagonistic effects. For example, 95% posterior credible intervals can be utilized to conclude whether the estimated synergy between two drugs can be considered different from zero, i.e. whether this particular pair of drugs is of interest for further study. This can provide a useful alternative to statistical hypothesis testing, if the prior distributions are chosen sensibly.

Acknowledgements

*A. Cremaschi and L. Rønneberg should be considered joint first authors. The authors acknowledge funding by the University of Oslo, Faculty for Mathematics and Natural Sciences and by the UiO Convergence Grant PerCaThe, the Research Council of Norway (NFR Centre of Excellence BigInsight, No. 237718), The Norwegian Cancer Society and The Regional Health Authority for South-Eastern Norway. The authors would like to thank Kjetil Taskén for important discussions on the topic of drug-drug combination experiments and synergy estimation.

References

Bliss CI (1939). "*The toxicity of poisons applied jointly.*" *Annals of applied biology*, **26**(3), 585–615.

Cremaschi A, Frigessi A, Taskén K, Zucknick M (2019). "A Bayesian approach for the study of synergistic interaction effects in *in-vitro* drug combination experiments." *arXiv preprint arXiv:1904.04901*.

de Boor C (2001). "A Practical Guide to Splines (Applied Mathematical Sciences Vol. 27)."

Di Veroli GY, Fornari C, Wang D, Mollard S, Bramhall JL, Richards FM, Jodrell DI (2016). "Combenefit: an interactive platform for the analysis and visualization of drug combinations." *Bioinformatics*, **32**(18), 2866–2868. ISSN 1367-4803.

Eddelbuettel D, Emerson JW, Kane MJ (2019). *BH: Boost C++ Header Files.* R package version 1.69.0-1.

Eddelbuettel D, François R (2011). "Rcpp: Seamless R and C++ Integration." *Journal of Statistical Software*, **40**(8), 1–18.

Eddelbuettel D, Sanderson C (2014). "RcppArmadillo: Accelerating R with high-performance C++ linear algebra." *Computational Statistics and Data Analysis*, **71**, 1054–1063.

Eilers PHC, Marx BD (2010). "Splines, knots, and penalties." *Wiley Interdisciplinary Reviews: Computational Statistics*, **2**(6), 637–653.

Fouquier J, Guedj M (2015). "*Analysis of drug combinations: current methodological landscape.*" *Pharmacology research and perspectives*, **3**(3).

Gelman A, *et al.* (2006). "Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper)." *Bayesian analysis*, **1**(3), 515–534.

Greco WR, Bravo G, Parsons JC (1995). "The search for synergy: a critical review from a response surface perspective." *Pharmacological reviews*, **47**(2), 331–385.

Griffin JE, Stephens DA (2013). "Advances in Markov chain Monte Carlo." pp. 104–144.

He L, Kulesskiy E, Saarela J, Turunen L, Wennerberg K, Aittokallio T, Tang J (2018). *Methods for High-Throughput Drug Combination Screening and Synergy Scoring*, chapter 17, pp. 351–398. Springer New York.

Loewe S, Muischnek H (1926). "*Über Kombinationswirkungen.*" *Naunyn-Schmiedebergs Archiv für experimentelle Pathologie und Pharmakologie*, **114**, 313–326.

Mathews Griner LA, *et al.* (2014). "*High-throughput combinatorial screening identifies drugs that cooperate with ibrutinib to kill activated B-cell-like diffuse large B-cell lymphoma cells.*" *Proceedings of the National Academy of Sciences*, **111**(6), 2349–2354.

Meyer CT, Wooten DJ, Paudel BB, Bauer J, Hardeman KN, Westover D, Lovly CM, Harris LA, Tyson DR, Quaranta V (2019). "Quantifying Drug Combination Synergy along Potency and Efficacy Axes." *Cell systems*, **8**(2), 97–108.

Ritz C, Baty F, Streibig JC, Gerhard D (2015). "Dose-Response Analysis Using R." *PLOS ONE*, **10**(e0146021).

Vela CM, McBride A, Jaglowski SM, Andritsos LA (2016). "Ibrutinib for treatment of chronic lymphocytic leukemia." *American Journal of Health-System Pharmacy*, **73**(6), 367–375.

Williams CK, Rasmussen CE (2006). "Gaussian processes for machine learning." *the MIT Press*, **2**(3), 4.

Yadav B, Wennerberg K, Aittokallio T, Tang J (2015). "*Searching for Drug Synergy in Complex Dose-Response Landscapes Using an Interaction Potency Model.*" *Computational and Structural Biotechnology Journal*, **13**, 504–513.

Yadav B, *et al.* (2014). "*Quantitative scoring of differential drug sensitivity for individually optimized anticancer therapies.*" *Scientific reports*, **4**(5193).

Affiliation:

Andrea Cremaschi
Department of Cancer Immunology
Institute of Cancer Research
Oslo University Hospital Montebello
0310 Oslo, Norway

Andrea Cremaschi, Leiv Rønneberg, Arnoldo Frigessi, Manuela Zucknick
Oslo Centre for Biostatistics and Epidemiology
Department of Biostatistics
Institute of Basic Medical Sciences
University of Oslo
P.O. Box 1122 Blindern
0317 Oslo, Norway
andrea.cremaschi@medisin.uio.no
leiv.ronneberg@medisin.uio.no
arnoldo.frigessi@medisin.uio.no
manuela.zucknick@medisin.uio.no