

# Predicting Significant Coronary Disease

## Unraveling the Complexity of Coronary Health through Bayesian Logistic Regression

anonymous

### 1 Introduction

Cardiovascular diseases, particularly coronary artery disease (CAD), remain a leading cause of mortality worldwide. In the pursuit of effective diagnosis and intervention, cardiac catheterization has emerged as a procedure for evaluating and addressing coronary artery health.

A reduction in artery diameter by at least 75% is considered a significant reduction in blood flow that puts downstream heart muscle at risk of ischemia. Such a reduction is defined as significant coronary disease. This report explores the complexities surrounding the cardiac catheterization process, with a specific focus on predicting the presence of significant coronary disease through the utilization of Bayesian models.

Duke University Medical Center's Cardiovascular Disease Databank provides a data set containing information about patients that has been diagnosed at Duke University Medical Center. The patients were referred to Duke for chest pain and diagnosed accordingly. If the patient was diagnosed with significant coronary disease, the blockade can be of either all three of the coronary vessels, or only of the left main coronary artery. Both conditions are associated with a high risk for mortality.

The scope of this report is to implement two Bayesian models able to predict the probability of significant coronary disease among patients. The two models are based on Bayesian logistic regression, but one is linear and one is nonlinear. After implementation the models are evaluated and compared to achieve the best possible result.

In section 2 the data used in this report is described. Section 3 explains the choice of mathematical model used in this report. In section 4 the two Stan models are implemented, and section 5 evaluates and compares the implemented models. Finally, section 6 & 7 includes a final discussion and a reflection on the lessons learned during the project.

## 2 Data Description

The “cath” dataset used in this report is obtained from Duke University Cardiovascular Disease Databank (“HBioStat Data” 2023). It encapsulates a collection of 6 variables (Table 1) that are closely related to cardiovascular health.

Table 1: The cath dataset.

sex	age	cad_dur	choleste	sigdz	tvdln
0	73	132	268	1	1
0	68	85	120	1	1
0	54	45	NA	1	0
1	58	86	245	0	0
1	56	7	269	0	0
0	64	0	NA	1	0

The dataset consists of four explanatory variables (*sex*, *age*, *cad\_dur*, *choleste*) and two response variables (*sigdz*, *tvdln*) that provide an overview on patient demographics, clinical indicators, and critical outcomes related to coronary artery disease:

- **Sex** (*sex*): Categorized as 0 for male and 1 for female, this variable represents the gender distribution within our dataset.
- **Age** (*age*): Representing the age of patients in years, this variable serves as a demographic feature.
- **Chest Pain Duration** (*cad\_dur*): The duration of chest pain symptoms in days.
- **Serum Cholesterol Level** (*choleste*): Measured in milligrams per deciliter, serum cholesterol levels are indicative of lipid metabolism and play a crucial role in cardiovascular health.
- **Significant Coronary Disease** (*sigdz*): A binary variable that captures the presence (1) or absence (0) of at least 75% blockage in one of the major coronary arteries.
- **Three Vessel Disease or Left Main Disease** (*tvdln*): Denoting the presence (1) or absence (0) of blockage in either all three coronary vessels or in the left main coronary artery.

The univariate distributions of these variables are visualized in Figure 1.

While constructing the Bayesian models to predict the probability of significant coronary disease, the report strives to utilize the correlation between the explanatory variables (*sex*, *age*, *cad\_dur*, *choleste*) and the desired response variable (*sigdz*). The *tvdln* variable is not

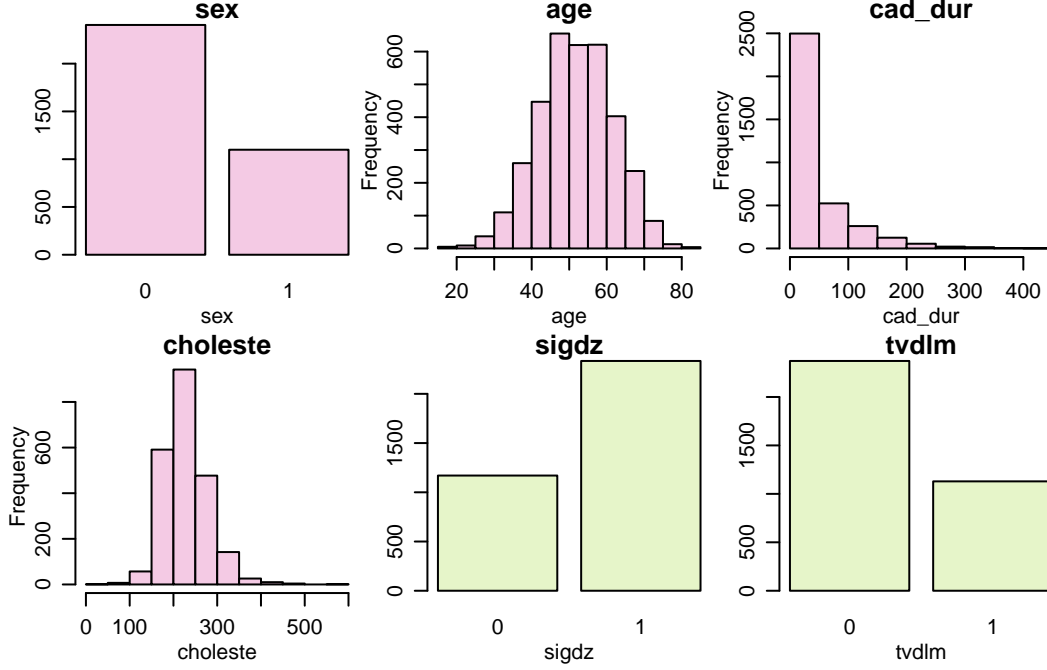


Figure 1: Distributions of the variables in the cath dataset.

relevant in this report as the main focus is to predict the probability of significant coronary disease, independent of the type of the blockade.

Before the analysis, the data is preprocessed by removing *tvdlm* column and all rows that contain missing values, as well as by scaling the continuous variables to zero mean and unit variance. After this, we are left with  $n = 2258$  observations. The pairwise correlations of variables are visualized in Figure 2. We can see that variables *sex* and *age* have the most significant bivariate correlation to the responsive variable *sigdz*.

### 3 Mathematical Model

In this analysis, we will construct two models for inferring the binary response variable, *sigdz* based on input explanatory variables. The first model is a generalized linear model (GLM), namely Bayesian logistic regression. The other model is a generalized additive mixed model (GAMM), which implements Bayesian logistic regression with nonlinear transformations on the input variables. These models will be referred to as linear and nonlinear model, respectively.

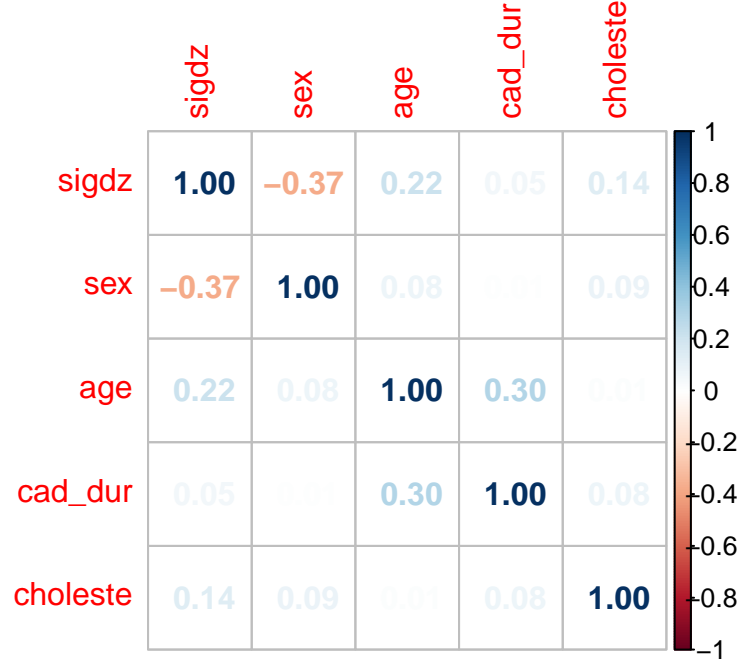


Figure 2: Correlation plot of 4 explanatory variables and the response variable sigdz.

### 3.1 The generalized linear model and priors

Let  $y$  be the number of times the variable *sigdz* is realized to be 1 for one individual in the dataset, and let  $x$  be the explanatory variables for this outcome. Then, this number of successes for one individual follows a Binomial distribution

$$y \sim \binom{n}{y} \theta^y (1 - \theta)^{n-y},$$

where  $n$  is the number of observations for that specific individual and  $\theta = g^{-1}(\eta)$  ( $\eta = \alpha + x^T \beta$ ) is the probability of success (patient presenting with significant coronary disease). The inverse link function  $g^{-1}$  maps the output of the linear predictor  $\eta$  to a probability interval between 0 and 1.

For the binomial GLM, this project utilizes logit  $g(x) = \ln(\frac{x}{1-x})$  as a link function, which makes it a logistic regression model. As each individual occurs only once in the data,  $y$  can be directly presented as the binary response variable. Therefore, the likelihood of the response variable of one individual is reduced to Bernoulli distribution

$$y \sim \text{logit}^{-1}(\eta)^y (1 - \text{logit}^{-1}(\eta))^{1-y}.$$

The complete data likelihood is then a product of  $n = 2258$  likelihoods, with unshared probability of success.

As in Bayesian logistic regression the scope is to infer the distribution of the regression weights, namely the intercept  $\alpha$  and coefficients  $\beta = [\beta_1, \beta_2, \beta_3, \beta_4]^T$ , we define the prior to be Student's  $t$ -distribution

$$\begin{aligned}\alpha &\sim t_v(\mu, \sigma) \\ \beta_k &\sim t_v(\mu, \sigma), \quad k = 1, \dots, 4\end{aligned}$$

where  $v$  is the degrees of freedom,  $\mu$  is the location and  $\sigma$  is the scale.

The selection of prior distribution was done based on the nature of the data. Due to correlations, there is reason to believe that the parameters are not very close to zero, but most are still rather small than large. Therefore, as Student's  $t$ -distribution has heavy tails and larger scale compared to, for example, Gaussian distribution,  $t$ -distribution is a suitable choice of prior for this purpose. The parameters of the prior were defined to be

$$v = 3, \quad \mu = 0, \quad \sigma = 2.5,$$

as the coefficients can be positive or negative.

Finally, the joint posterior distribution that is simulated using Hamiltonian Monte Carlo (HMC) is proportional to the product of likelihood and prior distributions:

$$p(\alpha, \beta | \mathbf{x}, \mathbf{y}) \propto t_v(\alpha | \mu, \sigma) \times \prod_{k=1}^4 t_v(\beta_k | \mu, \sigma) \times \prod_{i=1}^n \text{logit}^{-1}(\eta_i)^{y_i} (1 - \text{logit}^{-1}(\eta_i))^{1-y_i}$$

### 3.2 The generalized additive nonlinear model

The additive nonlinear model on the other hand, is a generalized additive mixed model which combines multiple functions in a way that is not strictly linear. This allows for a more flexible relationship between the explanatory variables and response variable  $y$ . In this report, the nonlinear model uses the same link function as the linear model, and so the nonlinear predictor for the logistic regression model can be written as:

$$\text{logit}(\theta) = \eta = \alpha + \sum_{k=1}^4 \beta_k f_k(x_k),$$

where  $f_k$  are nonlinear functions that transform the explanatory features individually. In this report, all functions of the continuous variables are smoothing functions. The  $f_{age}$  is simply

the variable itself, due to *sex* being binary variable. The smoothing functions utilize penalized splines, allowing the model to create a curved relationship between the features. The shape of the smoothing functions is estimated from the data and the penalty helps avoid overfitting. The smoothing function works by minimizing the sum of the model fit with the smoothness / penalty (here, thin plate regression splines (default)).

The likelihood of observation is the same as with the linear model, with the difference that a nonlinear transformation is applied to all input explanatory variables  $\mathbf{x}$ . Additionally, we utilize the same prior in both models to make them as comparable as possible. The posterior is again similar as with the linear model, but with the difference of nonlinear transformations on the input data.

## 4 Model Definitions and Implementation

The linear and nonlinear models are implemented as Stan code with the `rstanarm` package as described below. Both models were implemented with identical number of chains, draws and warm-up. The default values (`chains = 4`, `draws = 4000`, and `warmup = 2000`) were used in both cases.

### 4.1 Linear model

The linear model was implemented with the help of the `stan_glm` function from the `rstanarm` package. `stan_glm` is used to fit generalized linear models and performs a full Bayesian estimation with Markov Chain Monte Carlo (MCMC) estimation instead of maximum likelihood estimation and by adding priors to the GLM coefficients and intercept. By defining the model parameter `family = binomial(link = 'logit')` the model performs logistic regression.

The linear relationship between the response variable `sigdz` and the explanatory variables are defined with the help of the `formula` function and the prior for the regression coefficients and intercept with the help of the `student_t` function.

```
# Make response variable a factor
cath$sigdz <- as.factor(cath$sigdz)

# Formula
formula_linear <- formula(sigdz ~ sex + age + cad_dur + choleste)

# Prior
prior_linear <- student_t(df = 3, location = 0, scale = 2.5)

# The model
```

```

model_linear <- stan_glm(formula_linear, data = cath,
                        family = binomial(link = "logit"),
                        prior = prior_linear, prior_intercept = prior_linear,
                        QR=TRUE, refresh=0)

# saveRDS(model_linear, file = "./additional_files/model_linear.rds")
# model_linear <- readRDS("./additional_files/model_linear.rds")

```

## 4.2 Nonlinear model

The nonlinear model is similarly implemented utilizing `stan_gamm4` function from the `rstanarm` package and defined to be a logistic regression model with the help of the model parameter `family = binomial(link = 'logit')`. `stan_gamm4` fits a generalized additive mixed model, and performs Bayesian MCMC estimation instead of maximum likelihood estimation. In the same way as `stan_glm`, the model adds independent priors to the regression coefficients and intercept.

The nonlinear relationship between the response variable `sigdz` and the explanatory variables is again defined with the help of the `formula` function. This time, passing the smoothing function `s()` for all continuous explanatory variables, to allow for more complexity in the model, while simultaneously penalizing over-fitting of the model with the the thin plate regression splines smoothness.

The same priors are used as for the linear model for both the intercept and the regression coefficients.

```

# Formula
formula_nonlinear <- formula(sigdz ~ sex + s(age) + s(cad_dur) + s(choleste))

# Model definition
model_nonlinear <- stan_gamm4(
  formula_nonlinear, data = cath,
  family = binomial(link = "logit"),
  prior = prior_linear, prior_intercept = prior_linear,
  refresh = 0
)

# saveRDS(model_nonlinear, file = "./additional_files/model_nonlinear.rds")
# model_nonlinear <- readRDS("./additional_files/model_nonlinear.rds")

```

## 5 Model Evaluation

After fitting the models, multiple evaluation metrics such as split- $\hat{R}$ , effective sample size (ESS) and number of divergent transitions were used to assess the convergence of MCMC chains separately for each model. Additionally, we perform posterior predictive checks, assess the model performances as well as compare the models utilizing leave-one-out cross validation (LOO-CV). Finally, we perform prior sensitivity analysis for both models.

### 5.1 Convergence diagnostics & posterior predictive checks

For the linear model, the posterior predictive check and posterior distributions of the parameters with 95 % credible interval are visualized in Figure 3. As we can see, the model fits the data relatively well, with a bit of a variation around the probability interval endpoints.

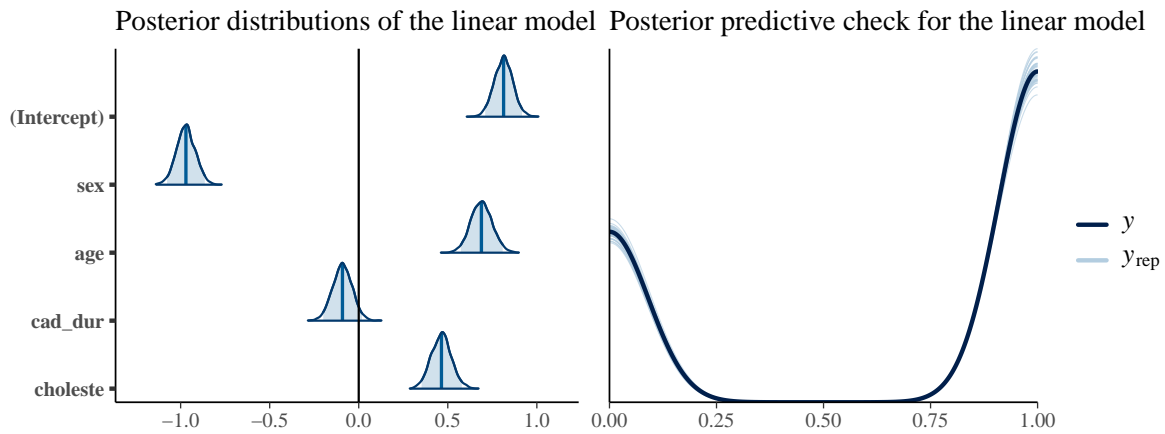


Figure 3: Posterior distributions and posterior predictive check for the linear model.

The convergence diagnostics for the linear model are summarized below.

Split-Rhat:

(Intercept)	sex	age	cad_dur	choleste
1.001	1.000	0.999	1.000	1.001
mean_PPD	log-posterior			
1.000	1.000			

Number of divergent transitions: 0

ESS ratio:



(Intercept)	sex	age	cad_dur	choleste
1.14375	1.07450	1.39525	1.46225	1.29375

Sd of ESS ratio: 0.164

The HMC chains have converged, as all split- $\hat{R}$  values are below 0.01. Additionally, there were no divergent transitions during convergence, and thus the HMC simulation is reliable. The ratio of the ESS to the true sample size is over 1 with all explanatory variables as well as the intercept. Although high ESS is generally a good thing, this also implies the MCMC samples may have a negative correlation.

For the nonlinear model, the posterior predictive check is visualized in Figure 4. Based on visual inspection, the nonlinear model fits the data as well as the linear model. Like with the linear model, the posterior has a bit of a variation around the probability interval endpoints.

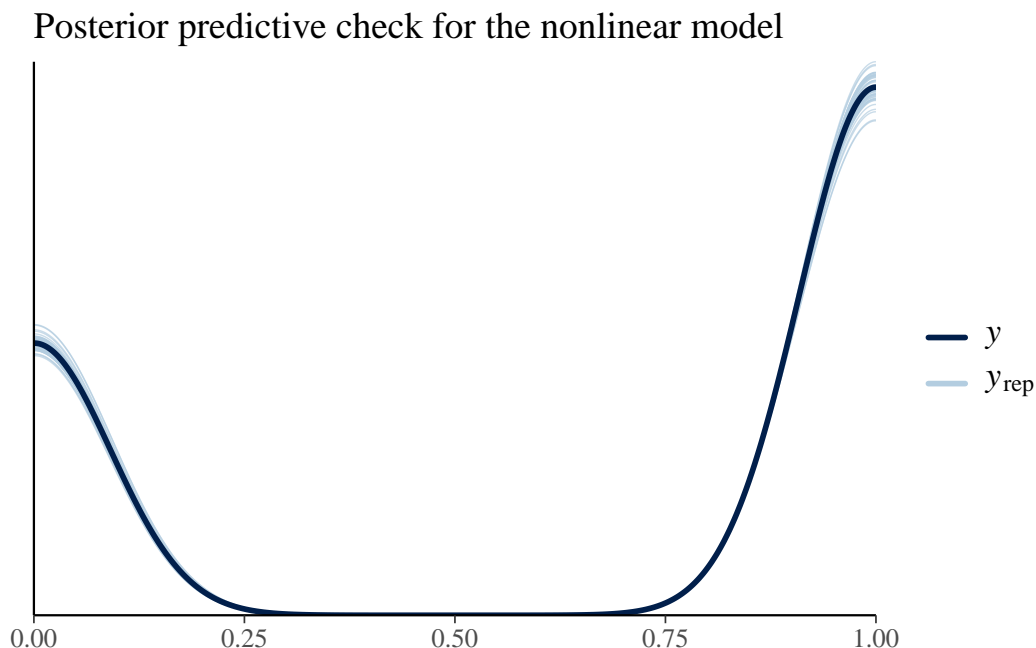


Figure 4: Posterior predictive check of the nonlinear model.

The convergence diagnostics for the nonlinear model are summarized below.

Split-Rhat of the first coefficients:

(Intercept)	sex	s(age).1	s(age).2	s(age).3	s(age).4
0.999	1.000	1.001	1.002	1.001	1.000

```
All split-Rhat <= 0.01: TRUE
```

```
Number of divergent transitions: 0
```

```
ESS ratio of the first coefficients:
```

(Intercept)	sex	s(age).1	s(age).2	s(age).3	s(age).4
1.273	1.044	0.907	0.798	0.410	1.006

```
Mean of ESS ratio: 0.784
```

```
Sd of ESS ratio: 0.339
```

Also the MCMC chains for this model have converged, as the split- $\hat{R}$  is less than 0.01 for all coefficients. As with the linear model, there were no divergent transitions during convergence, and thus the HMC simulation is reliable. On the other hand, the ESS ratio is on average less than 1, so a bit lower than with the linear model. However, the standard deviation of the ESS ratios is over twice larger than with the linear model. Low ESS ratio is not good in the model sense, but nevertheless the nonlinear has multiple coefficients to balance out the overall effect.

## 5.2 Model comparison using LOO-CV

To compare the performance of the linear model and nonlinear model to each other as well as to a baseline model, we will compute the expected log-densities of the predictive distributions (ELPD) by applying Pareto smoothed LOO-CV (PSIS-LOO) to the models. The baseline model is simply a logistic regression model without any explanatory variables and with a unit coefficient.

```
# Baseline model
model_baseline <- update(model_linear, formula = sigdz ~ 1, QR = FALSE)
```

The results of the PSIS-LOO for each model are presented below.

Linear model:

Computed from 4000 by 2258 log-likelihood matrix

	Estimate	SE
elpd_loo	-1177.6	24.4
p_loo	5.4	0.2
looic	2355.2	48.8

-----

Monte Carlo SE of elpd\_loo is 0.0.

All Pareto k estimates are good ( $k < 0.5$ ).  
See `help('pareto-k-diagnostic')` for details.

Nonlinear model:

Computed from 4000 by 2258 log-likelihood matrix

	Estimate	SE
elpd_loo	-1170.5	24.3
p_loo	11.6	0.7
looic	2341.0	48.6

-----

Monte Carlo SE of elpd\_loo is 0.1.

All Pareto k estimates are good ( $k < 0.5$ ).  
See `help('pareto-k-diagnostic')` for details.

Baseline model:

Computed from 4000 by 2258 log-likelihood matrix

	Estimate	SE
elpd_loo	-1448.7	14.9
p_loo	1.1	0.0
looic	2897.5	29.9

-----

Monte Carlo SE of elpd\_loo is 0.0.

All Pareto k estimates are good ( $k < 0.5$ ).  
See `help('pareto-k-diagnostic')` for details.

For each model, all Pareto  $k$  estimates are  $\leq 0.5$ , which implies the importance sampling gives a reliable estimate on the computed ELPDs. Additionally, for linear and nonlinear model the  $p_{loo}$  is less than the number of parameters in the respective model, and thus the model specifications seem to be reasonable.

To assess the model ELPDs with respect to each other, we compute the comparison between the results of PSIS-LOO:

	elpd_diff	se_diff
model_nonlinear	0.0	0.0
model_linear	-7.1	3.4
model_baseline	-278.2	21.8

It indeed seems that the predictive performances of both linear and nonlinear models are better compared to the baseline model. Additionally, the scale of the difference of standard errors ( $se\_diff$ ) of the baseline model and nonlinear model is smaller than the difference of the ELPDs of the nonlinear and baseline model, which implies the difference in predictive log-densities between these models is not simply explained by the variance. Although the difference in ELPDs of the linear and nonlinear models is small, the nonlinear model slightly outperforms the linear model.

### 5.3 Posterior predictive performance: Classification accuracy

To estimate the generalization error, i.e. the generalization and performance of models on unseen data, we compute LOO-balanced classification accuracies for the linear and nonlinear models. The results are summarized in Table 2. The classification accuracy is simply the fraction of correctly classified observations. The balanced classification accuracy on the other hand takes into account true positive rate (sensitivity) and true negative rate (specificity). The latter accounts for the imbalance in our data and is therefore more accurate estimate on the generalization error. Estimating generalization error is highly important for practical usage of the model, as it would be centered around predicting

Table 2: Imbalanced and balanced classification accuracies of the linear and nonlinear models.

	classification_accuracy	balanced_classification_accuracy
linear_model	0.75	0.69
nonlinear_model	0.76	0.69

We can see that the performances of both models are nearly equal. The classification accuracy is slightly better for the nonlinear model, but the balanced accuracy is the same for both

models. The the balanced classification accuracy is not very high, but nevertheless, both models outperform a random classifier.

To further compare the predictive performances of the models, calibration plots of both linear and nonlinear models are visualized in Figure 5. From these plots we can see that both models are calibrated on approximately the interval of  $[0.25, 0.5]$  of predicted probabilities. For the top 10% there are more predictions than observations, and for the bottom 20% there are more observations than predictions.

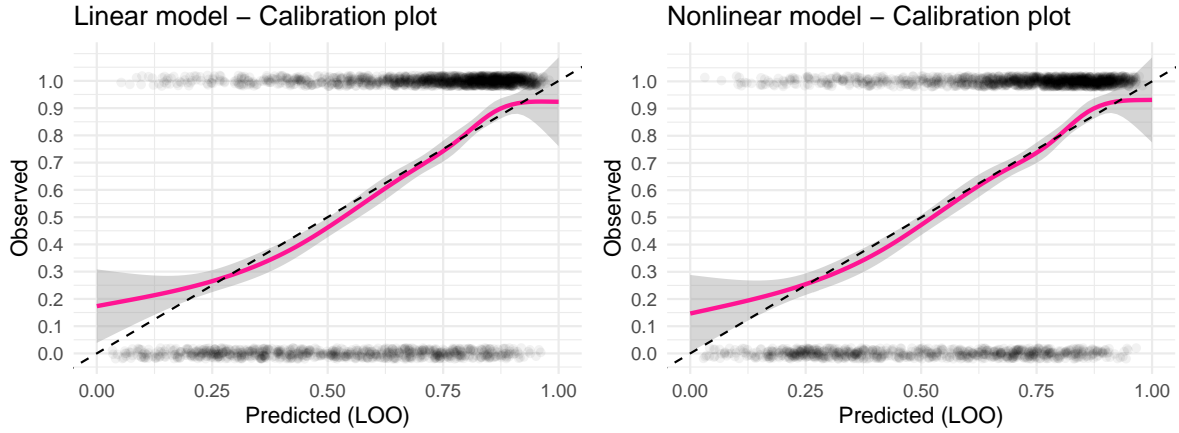


Figure 5: Calibration plots of the linear and nonlinear models.

## 5.4 Prior sensitivity analysis

For the prior sensitivity analysis, four different prior distributions were tested, namely a flat prior, a normal distribution with an excessively large location, a normal distribution with an excessively large scale and the default priors of the `stan_glm` and `stan_gamm4` functions.

The default priors

$$\alpha \sim \mathcal{N}(0, 2.5/sd(x))$$

$$\beta \sim \mathcal{N}(0, 2.5)$$

The large location and large scale normal priors

$$\theta \sim \mathcal{N}(100, 0)$$

The flat prior

$$\theta \sim C$$

where  $sd(x)$  is the standard deviation of  $x$ , and  $C$  is a constant value.

The linear and nonlinear model were run again with the new sets of priors and the outputs were compared to each other.

The priors were set up and the models run again (with otherwise same parameters as in the sections describing the models). The default priors were not separately defined, rather the defaults were passed inside the model function.

```
# normal prior, large scale N(0, 100)
normal_prior_sd <- normal(0, scale = 100, autoscale = FALSE)

# normal prior, large location N(100, 0)
normal_prior_loc <- normal(100, scale = 2.5, autoscale = FALSE)

# flat prior
flat_prior <- NULL
```

The posteriors of the linear model with the new sets of priors are plotted in `?@fig-prior-sensitivity-plots`.

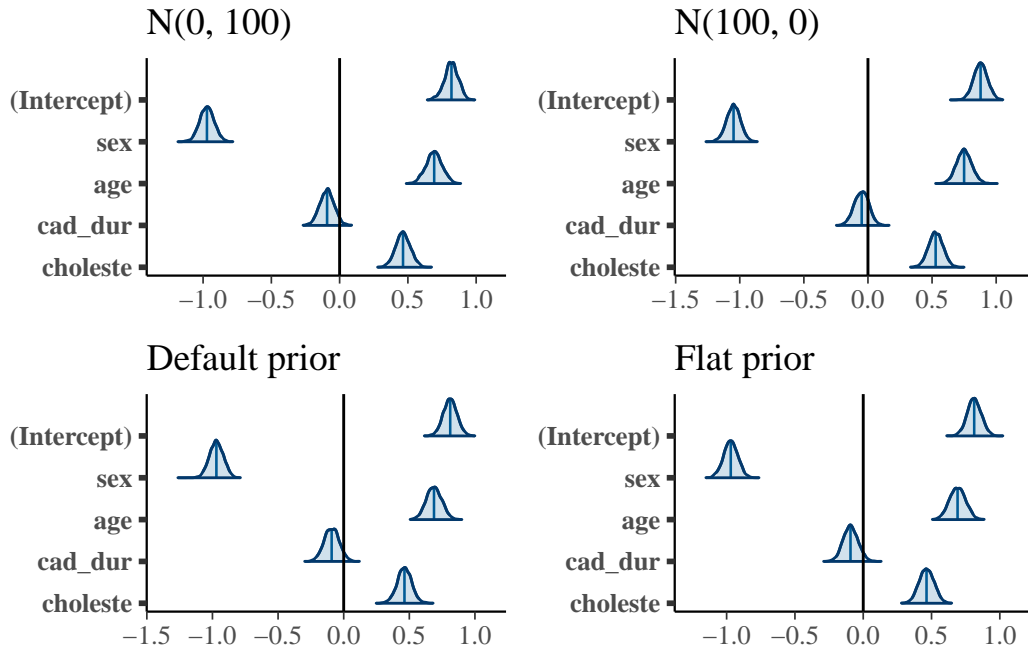


Figure 6: Prior sensitivity analysis on the linear model - Posterior distributions.

The posterior distributions look fairly similar despite the different priors. This indicates that the model isn't too sensitive to the prior selection.

Due to the nature of the nonlinear model, and the large number of parameters, the sensitivity analysis is summarized with the help of a posterior predictive check.

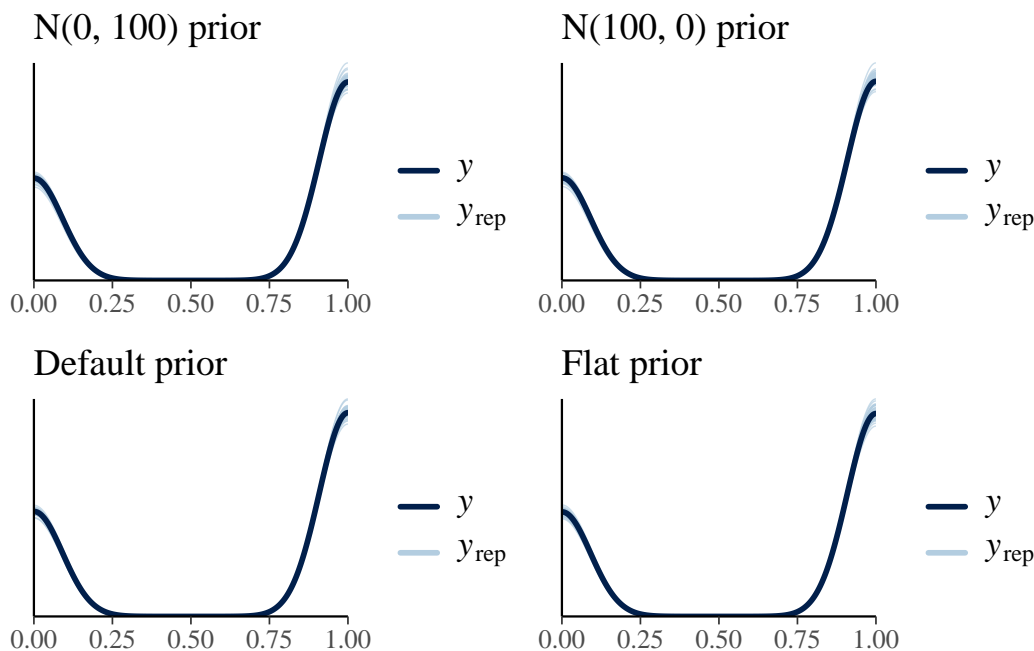


Figure 7: Prior sensitivity analysis on the nonlinear model - Posterior predictive checks.

The posterior predictive plots look fairly similar, which indicate a robust model.

Further, the divergent transitions were studied:

```
div_default <- sum(subset(nuts_params(nonlinear_model_default_prior), Parameter == "divergent_"))
div_flat <- sum(subset(nuts_params(nonlinear_flat_prior), Parameter == "divergent_"))
div_normal_loc <- sum(subset(nuts_params(nonlinear_normal_loc_prior), Parameter == "divergent_"))
div_normal_sd <- sum(subset(nuts_params(nonlinear_normal_sd_prior), Parameter == "divergent_"))
prior_sens_divergents <- data.frame(diversions = c(div_default, div_flat, div_normal_loc, div_normal_sd))
rownames(prior_sens_divergents) <- c("Default prior", "Flat prior", "N(100, 0) prior", "N(0, 100) prior")
kable(prior_sens_divergents)
```

	diversions
Default prior	0
Flat prior	0
$N(100, 0)$ prior	0
$N(0, 100)$ prior	1

## 6 Discussion

The scope of this report was to implement two Bayesian models able to predict the probability of significant coronary disease among patients. The models were implemented Bayesian logistic regression with both a linear and a nonlinear approach. The results were two Stan models of equal balanced classification accuracy.

With classification accuracies of 0.75 (linear) and 0.76 (nonlinear), both models performed adequately when it came to predicting the pervallance of significant coronary disease. The results indicate that both models outperform the baseline model, with the nonlinear model slightly surpassing the linear model in terms of predictive performance. However, when considering the balanced classification accuracy, the performance of both models dropped slightly.

There is still room for improvement when it comes to the predictive performance of the models. The problem here may lie in the explanatory variables being too general when it comes to predicting certain diseases. To further improve the predictive performance of the model, new explanatory variables may have to be introduced. Family history, fasting glucose levels, and smoking status are all possible explanatory variables closely related to cardiovascular health that could help improve model performance (Marcovitz et al. 2005).

In conclusion, this analysis indicates that when it comes the “cath” data set, the linear and the non-linear model performs equally. As seen in the prior sensitivity analysis, the choice of priors does not have a significant impact on the posterior predictive performance of the models. The training stage includes large quantities of data, resulting in robust models that are not sensitive when it comes to choice of priors.

## 7 Lessons Learned

In retrospect, the biggest challenge that our group faced was finding suitable data to implement a Bayesian model on. Before finding the “cath” data set, we tried with multiple other data sets without success. We definitely learned not to underestimate the importance of data selection and preprocessing. Without qualitative data it is impossible to make qualitative predictions.

As a group we also got to dive deeper into the implementation of Bayesian logistic regression models, both linear and non-linear. A process that the group agreed boosted the learning



outcome of the course. We now got to implement from scratch what we had touched on during the lectures and assignments.

## 8 References

“HBIOStat Data.” 2023. <https://hbiostat.org/data/>.

Marcovitz, Pamela A., Hillary H. Tran, Barry A. Franklin, William W. O’Neill, Michael Yerkey, Judith Boura, Michael Kleerekoper, and Christine Z. Dickinson. 2005. “Usefulness of Bone Mineral Density to Predict Significant Coronary Artery Disease.” *The American Journal of Cardiology* 96 (8): 1059–63. <https://doi.org/https://doi.org/10.1016/j.amjcard.2005.06.034>.