POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

# Bayesian Models for early diagnosis of Metabolic Syndrome in healthy blood donors

Project Report in
Mathematical Engineering - Bayesian Statistics

**Farzin Nasiri, 10968409**
**Nicole Alessandra Negroni, 10622008**
**Alberto Pecora, 10715464**
**Paolo Trifilio, 10710662**
**Aurora Vido, 10742763**
**Tommaso Giovanni Volonteri, 10709977**

---

**Tutors:**
Simone Colombara
Prof. Ilenia Epifani

**Professor:**
Prof. Alessandra Guglielmi

**Academic year:**
2024-2025

**Abstract:** Metabolic syndrome poses significant health risks, making early identification in blood donors crucial for preventive care. This study develops a Bayesian model to predict the risk of metabolic syndrome using longitudinal data from AVIS donors and subsequently designs a classification system. We focus on five key physiological variables, defining a donor as "at risk" if at least three fall outside a predefined range. By leveraging a Bayesian approach, we iteratively update our models to enhance predictive accuracy. Our methodology includes data simulation, analysis of autocorrelation in the time-series dataset, and assessment of dependencies among target variables. Once the optimal model is identified, a classifier is built to flag high-risk donors. This clinical decision-support tool assists healthcare professionals by recommending further assessments only for those classified as "at risk," optimizing medical resource allocation and enabling timely interventions.

**Key-words:** Bayesian models, time series, longitudinal data, blood donors, metabolic syndrome, autocorrelation, classification

# Introduction

The research field is the onset of the metabolic syndrome. This term is used to describe a cluster of conditions that increase the risk of diseases related to insulin resistance, such as developing heart disease, stroke, and type 2 diabetes.

Metabolic syndrome occurs when at least three of the following five target variables fall inside specific ranges (see Section 1): glucose, triglycerides, blood pressure, waist circumference, and HDL cholesterol.
Early detection of this condition could enable timely interventions through lifestyle adjustments.

Metabolic syndrome affects 24.3% of the European population, and its prevalence increases with age. [1]
Lifestyle changes, such as maintaining a healthy diet, exercising, losing weight, and managing stress, are effective in preventing or managing the condition, according to Mohamed et al. (2023). [2]

The dataset comprises healthcare data on lifestyle habits and blood parameters, provided by AVIS Milano (Associazione Volontari Italiani del Sangue). It was collected between 2009 and 2023 from 4329 regular blood donors (individuals with at least two donations), including 3488 males and 841 females.

Starting from a univariate Gaussian hierarchical model (credits to Francesca Arrigoni [3]), we wanted to enhance it by addressing the following questions:

- Which are the relations between the target variables?
- How to change the likelihood from the current model?

Moreover, after finding the best model, we also aimed at building a classifier for our donors.

# 1. Dataset description

## 1.1. Data sources

The data used in this study was provided by AVIS Milano. It was retrieved from two different databases:

- the EMONET database, which contains details on donations and donors medical exams,
- the AVIS database, which includes donors lifestyle information.

The original dataset comprised 268251 donors' records collected since 1992, but the analysis was limited to donations between 2009 and September 2023, as lifestyle data were only available starting in 2009.

Furthermore, only regular donors — those with more than two donations during this period — were included, resulting in a final dataset of 4329 donors (3488 males and 841 females).

Since the dataset comes from the donation process, it is not representative of the general population. Indeed, to donate, candidates must meet certain criteria to protect both the health of the recipient and of the donor.
Italian donation rules are:

- all donors must be between 18 and 60 years, with the possibility of increasing the maximum age to 65 years and even 70 years if a doctor approves;
- all donors must weigh more than 50 kg;
- the systolic pressure must be under 180 mmHg;
- the diastolic pressure must be under 100 mmHg;
- the resting heart rate must be between 50 and 100 beats/min;
- the hemoglobin must be over 13.5 g/dL for male donors and over 12.5 g/dL for female donors.

Also there are rules regarding donation frequency:

- men and women not of reproductive age can donate every three months,
- women of reproductive age are allowed a maximum of two donations per year, with a minimum interval of three months.

## 1.2. Target variables

Metabolic syndrome refers to a group of conditions that often occur together, significantly raising the risk of developing heart disease, stroke, and type 2 diabetes.

To diagnose metabolic syndrome in a donor at least three of these variables must be within the specified range:

- high blood sugar (insulin resistance): fasting blood glucose levels exceeding 100 mg/dL;
- elevated triglycerides: triglyceride levels higher than 150 mg/dL;
- high blood pressure (hypertension): consistently elevated blood pressure readings above 130/85 mm Hg;
- excess abdominal fat: waist circumference greater than 40 inches (101.6 cm) for men or 35 inches (88.9 cm) for women;
- reduced HDL cholesterol: HDL cholesterol levels below 40 mg/dL for men and below 50 mg/dL for women.

| **Target variable** | **Critical range** |
|---|---|
| *Glucosio* | $>$ 100 mg/dL |
| *Trigliceridi* | $>$ 150 mg/dL |
| *PMAX* | $>$ 130/85 mmHg |
| *Circonferenza_vita* | $>$ 102 cm (males) |
| | $>$ 88 cm (females) |
| *Colesterolo_Hdl* | $<$ 40 mg/dL (males) |
| | $<$ 50 mg/dL (females) |

Table 1: Critical values for the target variables

## 1.3. Data structure

The dataset contains healthcare data from 4329 regular donors (at least two donations), including 3488 males and 841 females.

It has a longitudinal structure, with each donor identified by their *CAI* (Anagrafic Internal Code). The dataset includes 39 variables, which are classified into the following categories:

- Time-invariant variables (4):
  - sex (M/F),
  - date of birth (YYYY-MM-DD),
  - Rhesus factor (positive/negative),
  - blood type (0/A/B/AB).
- Time-variant variables (35): represented as time series, each associated with a vector of timestamps indicating when the measurement was recorded.
  - Categorical variables (3): smoking habits (Y/N), drinking habits (Y/N), physical activity (Y/N),
  - Numerical variables (32): blood tests results.

  Note that these vectors have different lengths, both because different donors donate blood at different times and because not all the variables are measured at every donation.

After flattening the dataset (see Section 2.1) we computed the age in years of the donor at each timestamp as an additional covariate. In particular, the variable *Data_di_nascita* was substituted by this new variable *Età*.

Therefore the number of variables remains 39.

## 2. Exploratory data analysis

As a first step, two pairs of identical observations were identified, and the duplicates were removed in both cases. Moreover, two variables (*Basofili* and *Basofili_percentuale*) contained no recorded data (null) and were therefore removed.

We then transformed the longitudinal dataset into a flattened format, addressing potential outliers and handling any missing values.

After the exploratory analysis, in order to satisfy the assumptions of the Bayesian model, the target variables were log-transformed and the numerical covariates were standardized.

### 2.1. Flattening the dataset

Given the longitudinal structure of the dataset, where time-variant variables were represented as time series, we first ensured that the number of observations for each variable aligned with the corresponding measurement dates. All time-variant variables were found to be consistent.

To facilitate data access, the longitudinal dataset was processed to flatten the time-variant variables, represented by pairs of columns containing timestamps and their corresponding measurements.

Each time-variant variable is now transformed into a long format, where each subject, date, and measurement forms a separate row. Observations were then grouped by subject (*CAI*) and date, and aggregated using appropriate methods (e.g., mean for numeric data or mode for categorical data).

More precisely, for every unique pair date-donor a row (observation) was created (see Tables 2 and 3):

- if two or more time-varying variables had a value recorded at the same date (like x1 and y1 at date A in the tables), they are inserted in the same row;
- if a time-varying variable had not been observed at a certain date (like y2 at date B), then in the row associated to that date an NA was added;
- time-invariant covariates, when not measured, were manually added according to different criteria (add the mean for numeric data and the mode for categorical data).

| CAI | Dates xx | Value xx | Dates yy | Value yy |
|---|---|---|---|---|
| Donor 1 | Date A,<br>Date B,<br>Date C | x1,<br>x2,<br>x3 | Date A,<br>Date C,<br>Date D | y1,<br>y2,<br>y3 |

Table 2: Longitudinal Dataset (4329 obs.)

| CAI | Dates | Value xx | Value yy |
|---|---|---|---|
| Donor 1 | Date A | x1 | y1 |
| Donor 1 | Date B | x2 | NA |
| Donor 1 | Date C | x3 | y2 |
| Donor 1 | Date D | NA | y3 |

Table 3: Flattened Dataset (104822 obs.)

Finally, the *Data* column is converted to a date format for further processing. A new variable *Età* was created measuring the difference in years between the donation timestamp and the birth date, as previously mentioned in Section 1.3.

The final flattened dataset contains 104822 observations, each identified by the unique *CAI-Date* pair, and includes 39 variables.

## 2.2. Implausible values

Through visual inspection, we checked for negative values and adjusted signs where necessary. This process resulted in a single adjustment: the variable *Alanina* had a negative value of -9, which was corrected to +9.

Among all variables, we chose to identify and adjust potential outliers only for those with known acceptable ranges, such as *Altezza, Peso, Circonferenza_ vita, PMAX,* and *Polso*.
Other variables were left unmodified, as they were automatically generated during blood tests.

This correction procedure varied for each variable, as their acceptable ranges differ:

- to correct *Altezza*, implausible values were replaced with the mode, given that height remains relatively constant over time.
- for *Peso* and *Circonferenza_ vita*, a moving window approach was used: for every vector we took a sliding window of three elements, in which if a peak or a drop occurred with respect to the median of that window (see Table 4), then the implausible value was swapped with the median of that window. The thresholds were chosen based on a preliminary sensitivity analysis.

| | |
|---|---|
| *Peso* | $\geq$ 50 kg |
| *Circonferenza_ vita* | $\geq$ 40 cm |

Table 4: Non-acceptable peak

- for *PMAX* and *Polso* we kept the values within the ranges in Table 5 ([3]) and replaced any peaks or drops with the median of the entire time-series.

| | |
|---|---|
| *Polso* | 30-180 beats/min |
| *PMAX* | 60-200 mmHg |

Table 5: Acceptable range

## 2.3. Filling of NA values

The resulting dataset contained many missing values:

- "original" NA values, that is, the ones that were in the original dataset (not recorded data)
- "flattened" NA values, that is, all the NA values that were a result of the flattening operation (see Table 3).

To deal with this high number of NA values, we decided to do a NA filling operation.

We proceeded as follows: based on the donation frequency rules, we assumed that if two dates, A and B, were less than 15 days apart, then they referred to the same donation event. Therefore, any missing value (NA) in a time-varying variable on date A was assumed to be equal to the observed value on date B (see Tables 6, 7).

| CAI | Dates | Value xx | Value yy |
|---|---|---|---|
| Donor 1 | 01/01/2015 | x1 | y1 |
| Donor 1 | 02/01/2015 | x2 | NA |
| Donor 1 | 10/07/2015 | x3 | y2 |
| Donor 1 | 19/07/2015 | NA | y3 |
| Donor 1 | 25/09/2015 | NA | y4 |

Table 6: Flattened dataset before filling

| CAI | Dates | Value xx | Value yy |
|---|---|---|---|
| Donor 1 | 01/01/2015 | x1 | y1 |
| Donor 1 | 02/01/2015 | x2 | y1 |
| Donor 1 | 10/07/2015 | x3 | y2 |
| Donor 1 | 19/07/2015 | x3 | y3 |
| Donor 1 | 25/09/2015 | NA | y4 |

Table 7: Flattened dataset after filling

Due to the sparsity of the dates, the small window was insufficient to complete the dataset. However, we deemed 15 days a reasonable threshold, as increasing the interval would make it less plausible that values far apart in time correspond to the same event.

## 2.4. NA values in target variables

Next, we focused on the target variables, noting a significant number of missing values, particularly for *Circonferenza_ vita* and *Trigliceridi* (see Figure 1). This discrepancy arises because certain variables, such as *Glucosio* and *PMAX*, are measured at every donation, whereas others, like *Trigliceridi*, are only recorded annually. Additionally, variables such as *Circonferenza_ vita* have only been measured since 2019.

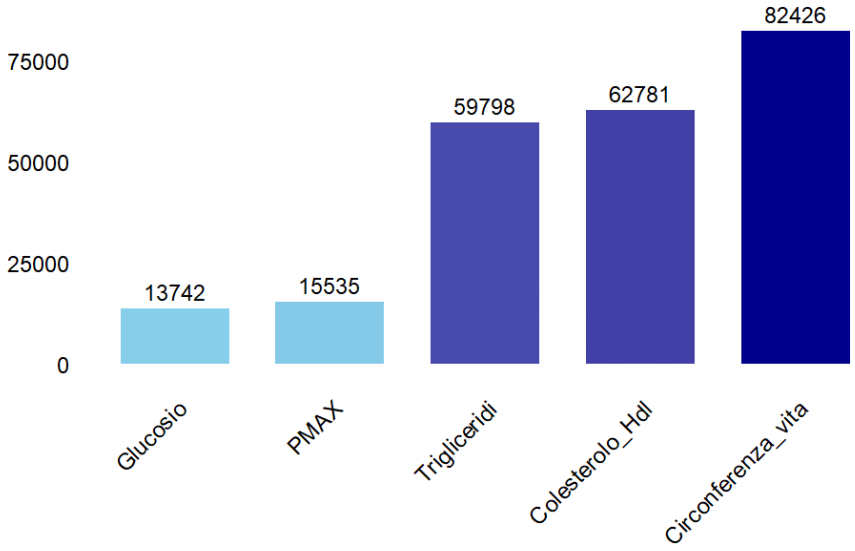


Figure 1: Target variables NAs (104822 obs.)

Since our final model will include the target variables simultaneously, we created this dataset in order to have a full matrix. Therefore, we had to cut all the observations that had NA values for the target variables.
This process led from the 104822 observation of the flattened dataset to 9152 values.

## 2.5. NA values in the covariates

We then analyzed the percentage of missing values for each covariate. As shown in Figure 2, the covariates *S_ gamma_ globuline, S_ beta_ 2_ globuline, S_ beta_ 1_ globuline, S_ alfa_ 2_ globuline, S_ alfa_ 1_ globuline*, and *Albumina* had an exceptionally high percentage of missing values and were therefore excluded. Additionally, we decided to exclude the covariates *Ferritina, Creatinina, Alanina_ aminotransferasi_ alt, Proteine_ totali* and *Ferro_ totale* as their missing values exceeded the 15% threshold we established.

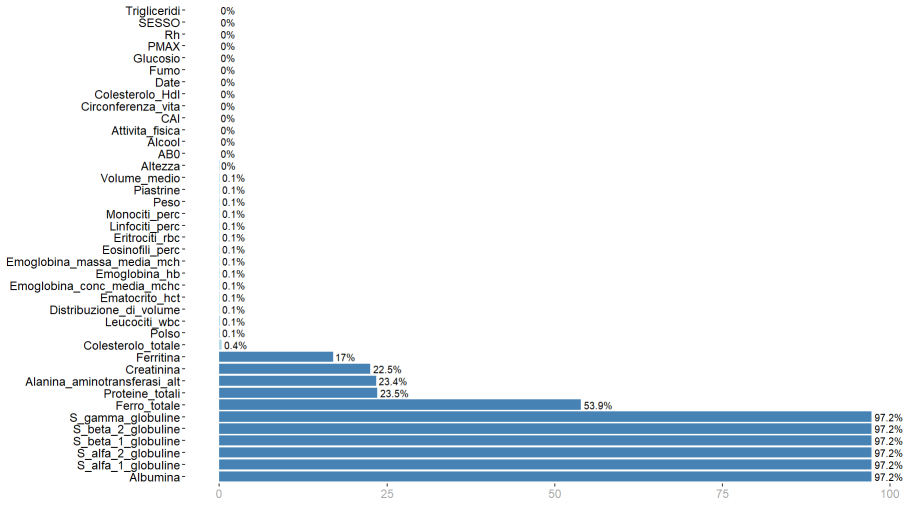


Figure 2: Variables NAs percentages (9152 obs.)

Then we cut the observations for which we still had NAs in the covariates.
We ultimately obtained 9081 observations across 28 variables.

# 3. Preliminary model

We first log-transformed the target variables and standardized the numerical covariates.

## 3.1. State-of-the-art model

The starting model, presented by Francesca Arrigoni in her thesis [3], is a univariate Gaussian hierarchical random effect model, where each target variable is modeled independently as

$$Y_i^{(k)} = X_i^{(k)}\beta^{(k)}, + \mathbf{1}b_i^{(k)} + \varepsilon_i^{(k)} \quad \varepsilon_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mathbf{0}, R_i^{(k)}) \quad \forall i = 1, ..., N, \quad \forall k = 1, ..., 5$$

where:

- $Y_i^{(k)}$: vector of repeated measurements for the $i$-th donor of the $k$-th target variable,
- $X_i^{(k)}$: observation matrix of the $i$-th donor for the $k$-th target variable.

The priors are:

- $\beta^{(k)}$: the vector of coefficients (without the intercept), with prior $\beta^{(k)} \overset{\textbf{iid}}{\sim} N(\mathbf{0}, \Sigma^{(k)})$,
- $b_i^{(k)}$: donor-specific random intercept, with priors $b_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mu, \eta^2)$, $\mu \sim (\mu_0, \sigma_{\mu_0}^2)$, $\eta^2 \sim Inv{-}Gamma(\alpha_0, \beta_0)$,
- $R_i^{(k)} = diag(r_i^{(k)})$, with prior $r_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mu_e, \sigma_e)$.

Note that the priors are exchangeable, so that the order in which we observe the data is not important.

## 3.2. Time-dependant model

Since the latter model does not take into account a possible dependence of the date on the blood donation, we modify it by adding the dependence on time $t$:

$$Y_{i,t}^{(k)} = X_{i,t}^{(k)}\beta^{(k)} + b_i^{(k)} + \varepsilon_{i,t}^{(k)}, \quad \varepsilon_{i,t}^{(k)} \overset{\textbf{iid}}{\sim} N(0, \sigma_{i,k}^2) \quad \forall i = 1, ..., N \quad \forall k = 1, \ldots, 5$$

where, in addition to the previous model:

- $Y_{i,t}^{(k)}$, value of the $k-$th target variable of donor $i$ at date $t$
- $X_{i,t}^{(k)}$: vector of covariates for the $i$-th donor at date $t$ for the $k$-th target variable.

The priors are:

- $\beta^{(k)} \overset{\textbf{iid}}{\sim} N(0, 5)$
- $b_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mu, \eta^2)$
- $\mu \sim N(0, 2)$
- $\eta^2 \sim Inv - Gamma(3, 2)$
- $\sigma_{i,k}^2 \overset{\textbf{iid}}{\sim} Inv - Gamma(3, 2)$.

In addition, $\sigma_{i,k}^2$, $b_i^{(k)}$, $\beta^{(k)}$ are assumed to be independent.

## 3.3. Convergence

The following traceplots represent the sampled values for parameters ($\beta$ and $\sigma$) across iterations of the MCMC simulation for each target variable.

The plots demonstrate that the MCMC chains have converged for all target variables and their corresponding parameters, as evidenced by their stationary behavior and homogeneity between the chains.

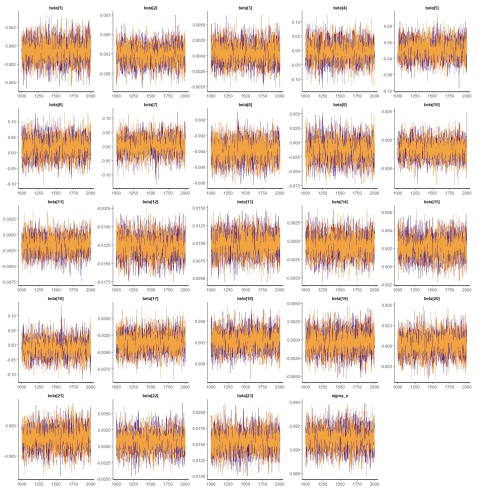

Figure 3: Glucose

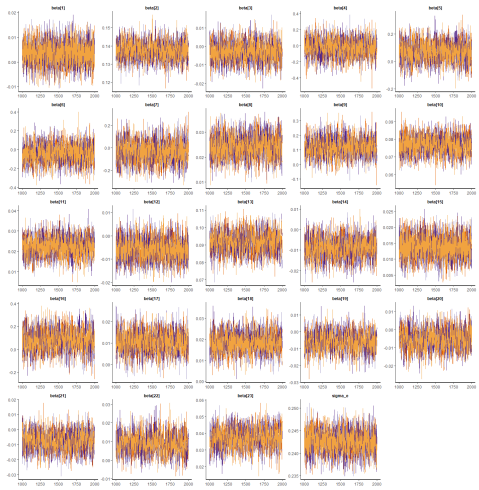

Figure 4: Triglycerides

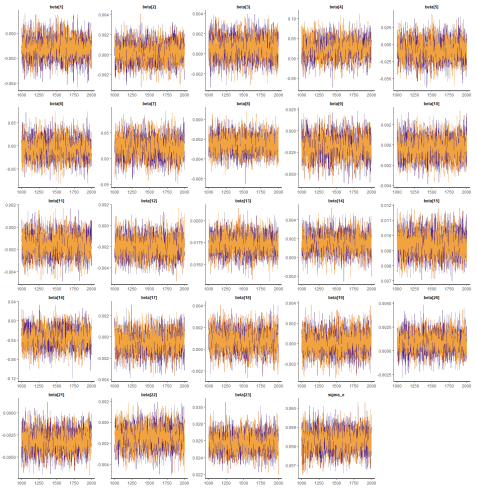

Figure 5: PMAX

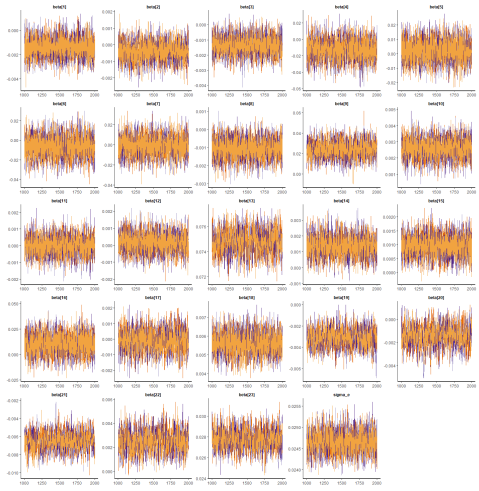

Figure 6: Waist Circumference

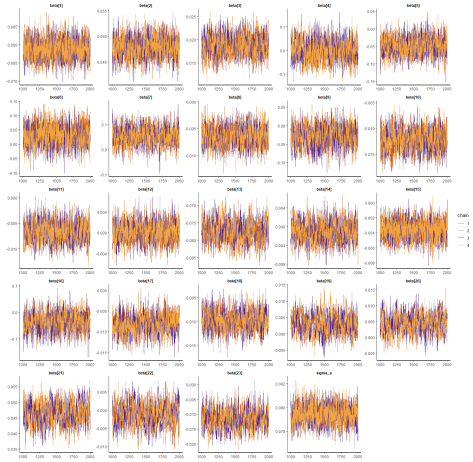

Figure 7: HDL Cholesterol

# 4. Model improvements

After analyzing the preliminary model, we explored potential improvements before proceeding to the final model. Three consequential approaches were considered:

- recover more observations,
- account for a possible autoregressive component,
- check for conditional dependencies between target variables.

## 4.1. MCMC imputation for missing values

The first approach involved addressing missing values in the target variables using Stan, in order to estimate the responses, based on the analysis of covariates, without any missing values.

Before actually performing MCMC imputation, we had to address missing values from the flattened data in the following way.

- Since the procedure of MCMC imputation required complete covariate data to estimate the responses, it was necessary to remove the missing values in the covariates. Therefore:
  - covariates with an NA percentage $\geq 15\%$[1] (see Figure 8) were removed, reducing the number of covariates from 34 to 19,
  - for the remaining covariates, rows containing any missing values were removed, reducing the number of observations from 104822 to 88853.

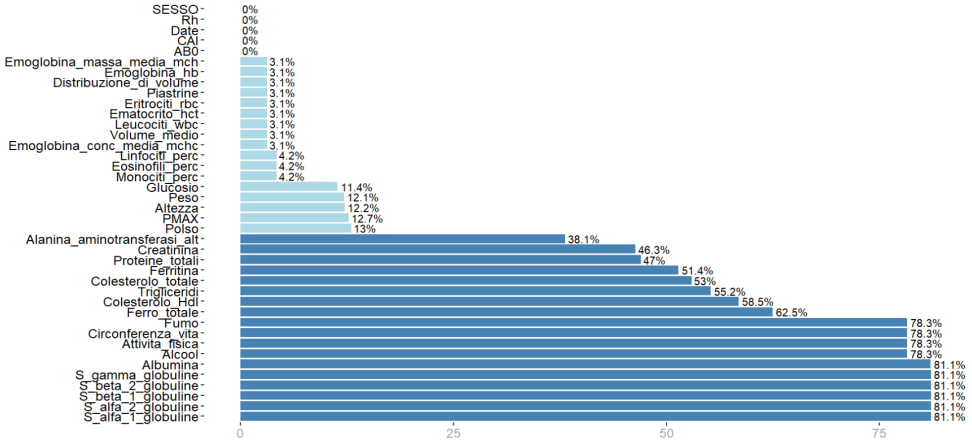


Figure 8: Variables NAs percentages (104822 obs.)

- Subsequently, since the NA values in the target variables were left unchanged, it was found that some donors had all NA values in at least one target variable. This presented a significant issue, as imputing missing values depends on learning from other data, which is not feasible when no values are available. Consequently, these donors were removed from the dataset, which now contains 88717 observations.

- We then revised the situation of the missing values in the target variables (see Figure 9) and decided to apply MCMC imputation only to *Trigliceridi*, *Colesterolo_Hdl*, and *Circonferenza_vita*. For *Glucosio* and *PMAX*, we removed the observations with missing values, as they were relatively sparse compared to the others. The number of observations was reduced from 88717 to 84955.

The final dataset now contains 84955 observations across 24 variables, of which 5 are target variables.

To do MCMC imputation, we therefore ran three different Stan models (see Appendix A.1) on *Trigliceridi, Circonferenza_vita, Colesterolo_Hdl*.

[1]Note that the threshold of 15% was chosen because the covariates with percentage between 10 and 15 were proven to be significant in the previous models, so we preferred to reduce the observations rather than to cut completely the variable.

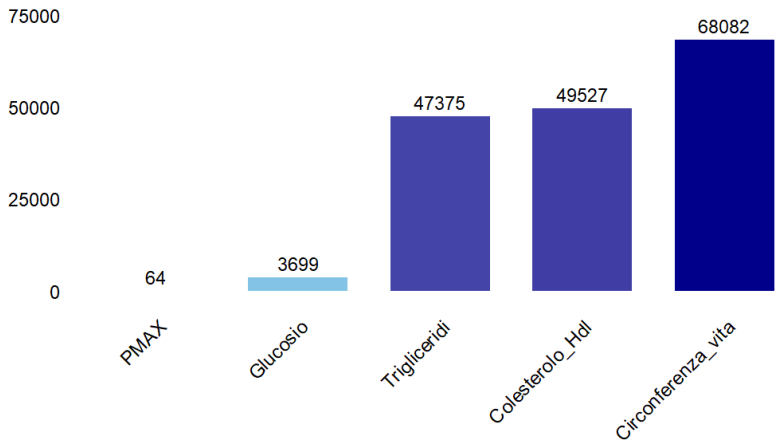


Figure 9: Target variables NAs percentages (88717 obs.)

## 4.2. Autoregressive model

The next step was trying to consider the presence of an autoregressive component in the target variables, so to implement for each target variable a model of the following type:

$$Y_{i,t}^{(k)} = X_{i,t}^{(k)}\beta^k + b_i^{(k)} + \Phi Y_{i,t-1}^{(k)} + \epsilon_{i,t}^{(k)} \quad \varepsilon_{i,t}^{(k)} \overset{\textbf{iid}}{\sim} N(0,\sigma_e) \quad i = 1,...,N \quad \forall k = 1,\dots,5$$

Where, in addition to the definitions in Section 3.2:

- $\beta^{(k)} \overset{\textbf{iid}}{\sim} N(0,5)$;
- $b_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mu_b,\eta)$, with $\mu_b \sim N(0,2)$ and $\eta \sim Inv-Gamma(3,2)$;
- $\sigma_e \sim Inv-Gamma(3,2)$;
- $\phi$ is the autoregressive coefficient and has prior $\phi \sim N(0,10)$.

How to define $Y_{i,t-1}^{(k)}$? Due to the fact that the timesteps were not homogeneous, we could not treat our process as a true AR model. Therefore, we opted for a threshold-based approach.

- The thresholds for *Colesterolo_Hdl* and *Trigliceridi* were chosen based on the literature [4] [5] [6], which suggested a large threshold (greater or equal than 2 years).
  However, comparing the Residual Sum of Squares:
  - RSS = 2402.384 for the 1-year threshold;
  - RSS = 2402.604 for the 2-year threshold;

  we chose the 1-year old threshold to be conservative, since the 2-year threshold performed slightly worse.
- For *Circonferenza_vita* a 1-year threshold was chosen too, since several studies consider intervals of 6 months to 2 years, depending on individuals' health conditions [7].
- Whilst for *Glucosio* and *PMAX* it is known that they are highly autocorrelated but most studies analyze this autocorrelation over a short time window, usually few minutes or hours [8] [9]. To explore autocorrelation for these targets, we considered the smallest possible interval between successive donations, namely 6 months (a 3-month window would not have included women, since they can donate every 6 months).

| Target variable | Threshold |
|---|---|
| *Glucosio* | 180 days |
| *Trigliceridi* | 365 days |
| *PMAX* | 180 days |
| *Circonferenza_vita* | 365 days |
| *Colesterolo_Hdl* | 365 days |

Table 8: Autoregression thresholds

The selected thresholds (Table 8) represent the maximum time intervals where meaningful correlations could persist between successive measurements. When running the Stan model (see Appendix A.2), for each instant:

- if the closest past observation is within the threshold, then we use it to build the autoregressive part,
- otherwise we set $Y_{i,t-1}^{(k)} = 0$.

The traceplots of the Stan models are shown below (left), along with the individual traceplots for each autoregressive coefficient (right), presented to assess both convergence and significance.

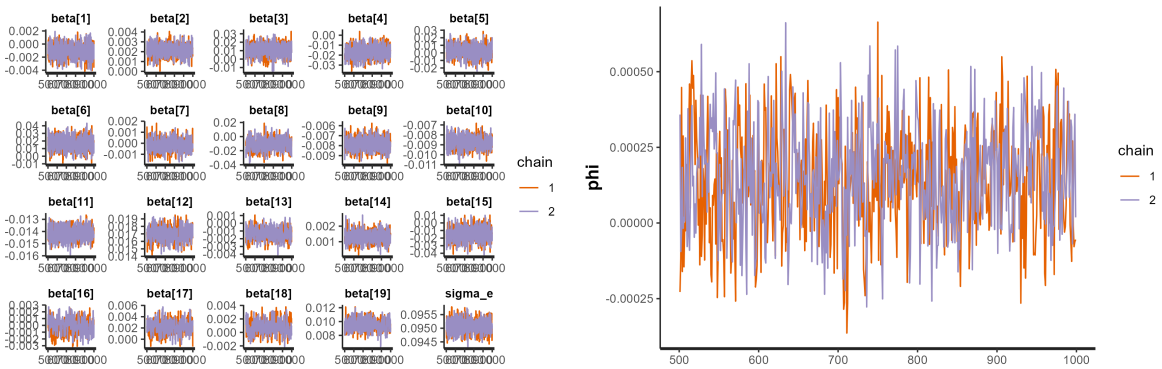


Figure 10: Glucose

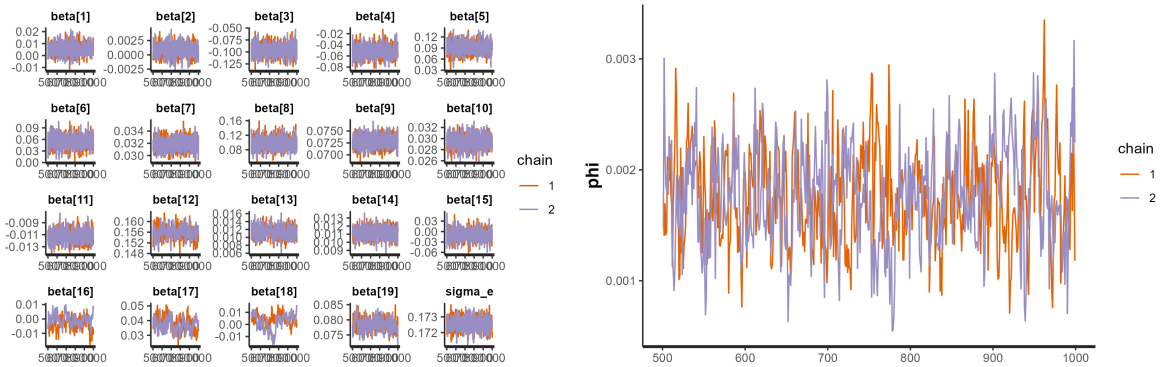


Figure 11: Triglycerides

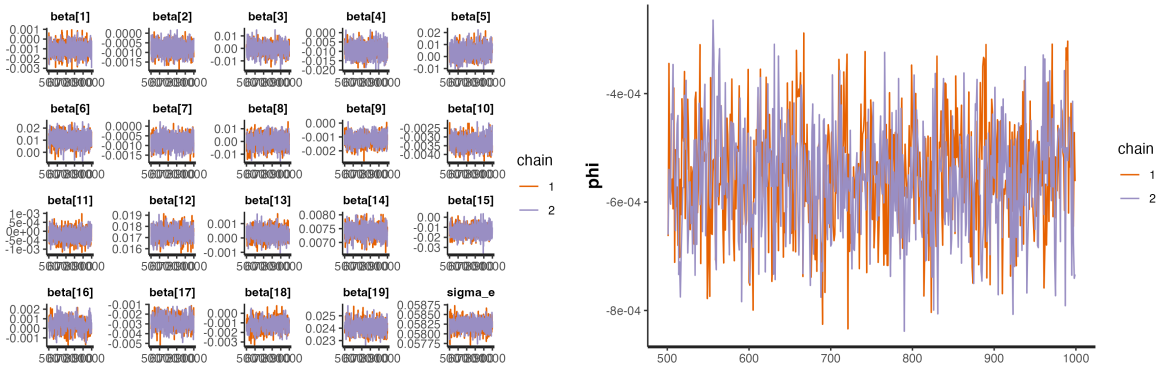


Figure 12: PMAX

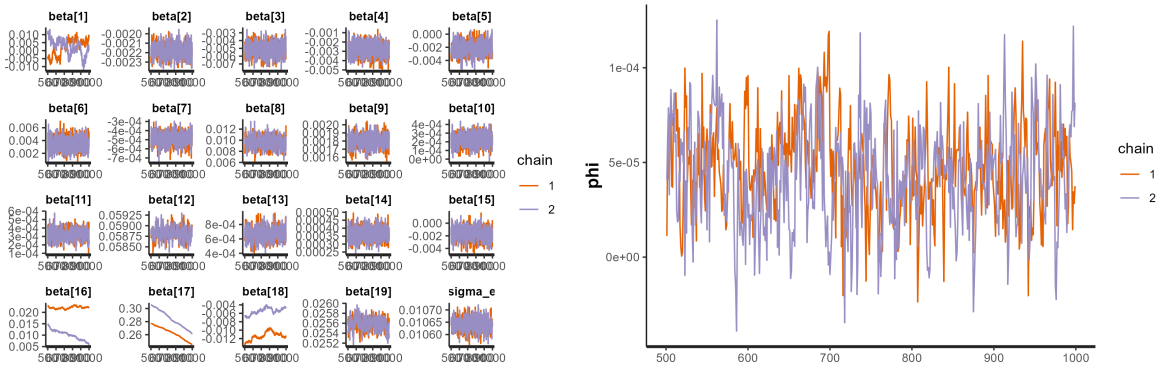


Figure 13: Waist Circumference

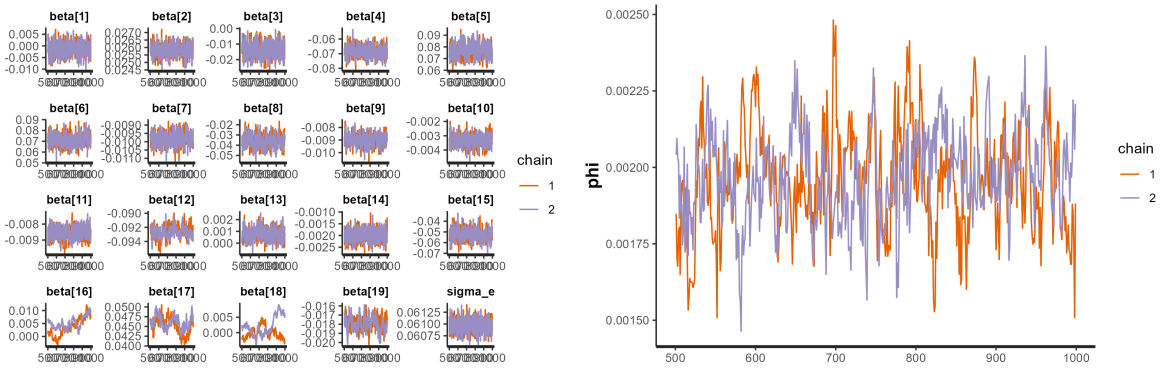


Figure 14: HDL Cholesterol

From the traceplots we observed that the categorical variables *Sesso*, *AB0* and *Rh*, had convergence problems in *Circonferenza_ Vita* and *HDL_ Cholesterol*. For now we only care about the autoregressive coefficient, but this problem will be addressed when building the final model.

In order to have a better interpretation, we analyzed the 95% credibility intervals for each target variable (see Table 9). It was observed that *Glucosio* and *Circonferenza_ vita* contained 0 inside the credibility interval.

| **Target variable** | **2,5%** | **97,5%** |
|---|---|---|
| *Glucosio* | -0.000186657 | 0.000485273 |
| *Trigliceridi* | 0.000911957 | 0.002636545 |
| *PMAX* | -0.000749735 | -0.000349545 |
| *Circonferenza_ vita* | -0.000007575 | 0.000009508 |
| *Colesterolo_ Hdl* | 0.001649848 | 0.002291993 |

Table 9: Autoregressive coefficients 95% credibility intervals

Therefore, we will assume the existence of a (very) small autoregressive coefficient only for the variables for which 0 is not included in the credibility interval: *Trigliceridi*, *PMAX* and *Colesterolo_ Hdl*.

## 4.3. Target dependencies

Finally, after recovering more observation and considering the autoregressive component, we checked for target dependencies. To do so, we used the R package BDgraph [10].

Given data, BDgraph identifies which variables are conditionally dependent on each other and creates a graph (network) that represents these dependencies. The nodes of the graph correspond to the variables, while the links represent the conditional dependencies between them.

In order to observe the "true" dependencies between the target variables, free from the influence of covariates and/or donor-specific coefficients, we chose to give as input the residuals obtained from the five autoregressive Stan models, instead of the original data.

The resulting graph is shown in Figure 15, where:

- nodes, numbered from 1 to 5, represent respectively the target variables *Colesterolo_ Hdl*, *Circonferenza_ vita*, *Glucosio*, *PMAX*, and *Trigliceridi*;
- links are represented only if the posterior probability of the dependency is greater than 0.5, by default.

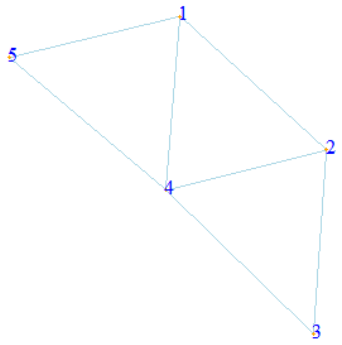


Figure 15: BDgraph target dependencies

Since the graph is undirected, no unique interpretation can be made. However, all target variables result to be connected, meaning that no target variable is completely independent from the others.

The joint distribution, of the target variables $Y_1, Y_2, Y_3, Y_4, Y_5$, used in the final model, is sequentially defined as:

$$\mathcal{L}(Y_1, Y_2, Y_3, Y_4, Y_5) = \mathcal{L}(Y_1)\mathcal{L}(Y_2|Y_1)\mathcal{L}(Y_3|Y_1, Y_2)\mathcal{L}(Y_4|Y_1, Y_2, Y_3)\mathcal{L}(Y_5|Y_1, Y_2, Y_3, Y_4).$$

To decide in which order to name the variables, we looked at the (mean) Bayesian $R^2$ [11] of the autoregressive models, computed as:

$$R^2 = \frac{\text{Var}(\hat{y})}{\text{Var}(\hat{y}) + \sigma_e^2}$$

Table 11 shows our final ordering (by decreasing $R^2$). The best performing variables will go first, while the worst performing ones will probably need to receive additional information from the other variables.

| Variable | BDgraph node | Target variable | Mean Bayesian $R^2$ |
|---|---|---|---|
| $Y_1$ | 2 | *Circonferenza_ vita* | 0.5175 |
| $Y_2$ | 5 | *Trigliceridi* | 0.4465 |
| $Y_3$ | 1 | *Colesterolo_ Hdl* | 0.4410 |
| $Y_4$ | 3 | *Glucosio* | 0.0475 |
| $Y_5$ | 4 | *PMAX* | 0.0450 |

Table 10: Final ordering of target variables according to the mean Bayesian $R^2$

# 5. Final model

Since the order of the target variables was defined, we could run the final Stan model, composed of five serial univariate models. In each model, to estimate the target variable, all preceding variables are used as additional covariates. This was obviously not the case for *Circonferenza vita*, since it was the first variable and therefore run alone.

In detail, the five models are: $\forall i = 1, \ldots, N$

$$Y_{i,t}^{(1)} = X_{i,t}^{(1)}\beta^{(1)} + b_i^{(1)} + \epsilon_{i,t}^{(1)}, \quad \varepsilon_{i,t}^{(1)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e)$$

$$Y_{i,t}^{(2)} = X_{i,t}^{(2)}\beta^{(2)} + b_i^{(2)} + \Phi Y_{i,t-1}^{(2)} + \gamma_1 Y_{i,t}^{(1)} + \epsilon_{i,t}^{(2)}, \quad \varepsilon_{i,t}^{(2)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e)$$

$$Y_{i,t}^{(3)} = X_{i,t}^{(3)}\beta^{(3)} + b_i^{(3)} + \Phi Y_{i,t-1}^{(3)} + \gamma_1 Y_{i,t}^{(1)} + \gamma_2 Y_{i,t}^{(2)} + \epsilon_{i,t}^{(3)}, \quad \varepsilon_{i,t}^{(3)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e)$$

$$Y_{i,t}^{(4)} = X_{i,t}^{(4)}\beta^{(4)} + b_i^{(4)} + \gamma_1 Y_{i,t}^{(1)} + \gamma_2 Y_{i,t}^{(2)} + \gamma_3 Y_{i,t}^{(3)} + \epsilon_{i,t}^{(4)}, \quad \varepsilon_{i,t}^{(4)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e)$$

$$Y_{i,t}^{(5)} = X_{i,t}^{(5)}\beta^{(5)} + b_i^{(5)} + \Phi Y_{i,t-1}^{(5)} + \gamma_1 Y_{i,t}^{(1)} + \gamma_2 Y_{i,t}^{(2)} + \gamma_3 Y_{i,t}^{(3)} + \gamma_4 Y_{i,t}^{(4)} + \epsilon_{i,t}^{(5)}, \quad \varepsilon_{i,t}^{(5)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e)$$

Or, in compact form,

$$Y_{i,t}^{(k)} = X_{i,t}^{(k)}\beta^{(k)} + b_i^{(k)} + \Phi Y_{i,t-1}^{(k)} + \gamma \mathbf{Y}_{\mathbf{i,t}}^{(\mathbf{k,prev})} + \epsilon_{i,t}^{(k)}, \quad \varepsilon_{i,t}^{(k)} \overset{\textbf{iid}}{\sim} N(0, \sigma_e) \quad \forall i = 1, \ldots, N \quad \forall k = 1, ..., 5$$

Where:

- $\beta^{(k)} \overset{\textbf{iid}}{\sim} N(0, 5)$;
- $b_i^{(k)} \overset{\textbf{iid}}{\sim} N(\mu_b, \eta)$, with $\mu_b \sim N(0, 2)$ and $\eta \sim Inv - Gamma(3, 2)$;
- $\sigma_e \sim Inv - Gamma(3, 2)$;
- $\phi$ is the autoregressive coefficient and has now a less vague prior $\phi \sim N(0, 1)$;
- $\mathbf{Y}_{\mathbf{i,t}}^{(\mathbf{k,prev})}$ is the vector of dependencies, and has length *k-1* (for every $Y^{(k)}$, $\mathbf{Y}_{\mathbf{i,t}}^{(\mathbf{k,prev})} = [Y_{i,t}^{j}]$, $j = 1, ..., k-1$). This term is absent from the model for *Circonferenza_ vita*.
- $\boldsymbol{\gamma} = [\gamma_j]$, $\gamma_j \overset{\textbf{iid}}{\sim} N(0, 5), j = 1, ..., k-1$ is the vector of dependence coefficients.

The dataset was split into training set and test set, in order to build a classifier: the last observation for each donor would go into the test set, while all the others in the training set.

In order to have good performances, at least 5 observations had to be in the training set, resulting in removing all donors with less than 6 total observations each.

## 5.1. Convergence

Some relevant observations about the models:

- The model for $Y_1$ (corresponding to *Circonferenza_ vita*) has been run twice, since the first time we observed that the categorical variables *Sesso, AB0* and *Rh* did not converge, and were therefore cut in the second run (see Figure 16). This was made in order not to propagate errors in the consequent models.
- The models for $Y_1$ and $Y_4$ (*Glucosio*) were run without the autoregressive coefficient since the analysis in Section 4.2 showed that the autoregressive coefficient was negligible for those variables.
- For the other target variables, both categorical covariates and the autoregressive coefficient were taken into consideration.

The traceplots below display the regression, autoregressive (for those that have it), and dependence coefficients for each model, arranged in the execution order.

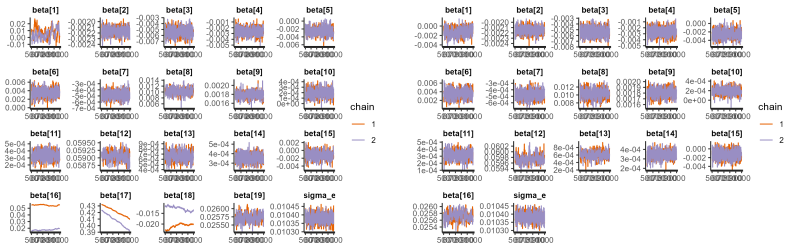

Figure 16: *Circonferenza_vita* coefficients: with (left) and without (right) the categorical variables

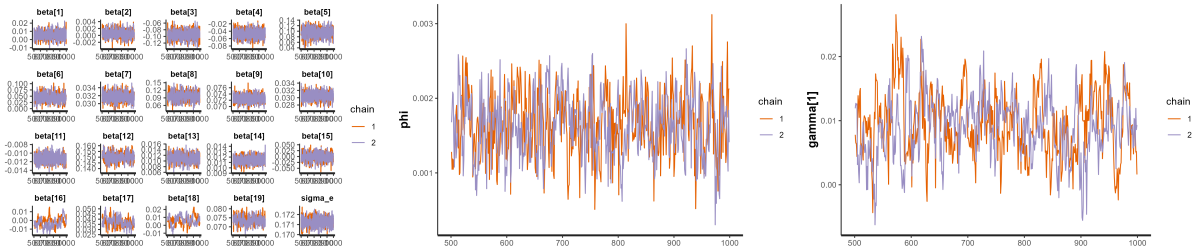

Figure 17: *Trigliceridi* coefficients: regression (left), autoregressive (center), dependence (right)

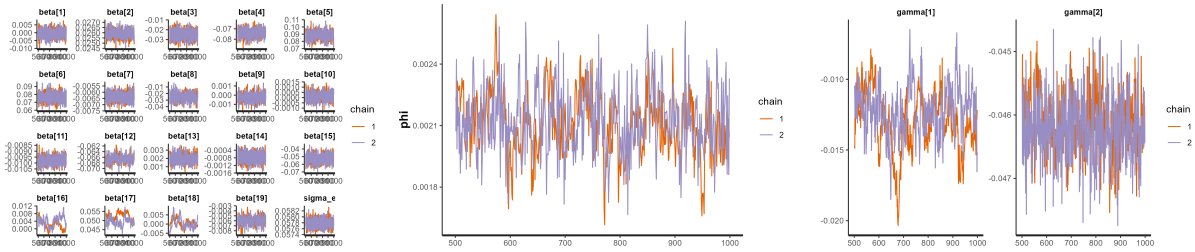

Figure 18: *Colesterolo_Hdl* coefficients: regression (left), autoregressive (center), dependence (right)

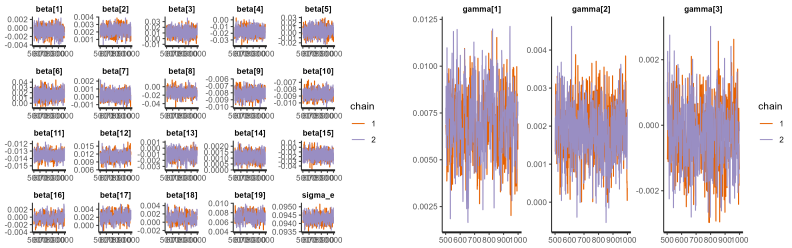

Figure 19: *Glucosio* coefficients: regression (left), dependence (right)

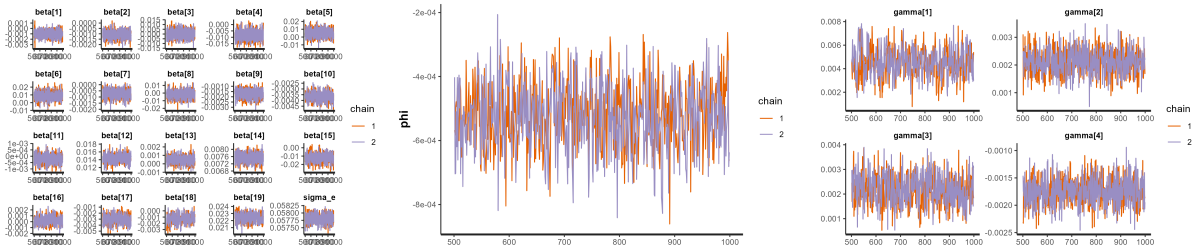

Figure 20: *PMAX* coefficients: regression (left), autoregressive (center), dependence (right)

| Target variable | Mean Bayesian $R^2$ Old | Mean Bayesian $R^2$ New | Improvement |
|---|---|---|---|
| *Circonferenza_vita* | 0.5175 | 0.5195 | 0.38% |
| *Trigliceridi* | 0.4465 | 0.4495 | 0.67% |
| *Colesterolo_Hdl* | 0.4410 | 0.5450 | 23.58% |
| *Glucosio* | 0.0475 | 0.0475 | 0% |
| *PMAX* | 0.0450 | 0.0455 | 1.11% |

Table 11: Mean Bayesian $R^2$ of each target variable in the final model

The final performances of our model can be seen in Table 11: all models improved their predictive capabilities, meaning that our approach is meaningful. The best result was achieved by *Colesterolo_Hdl*, which improved its performance by 23.58%. For the others, however, only a minimal improvement was obtained.

## 6. Classification

The final goal of the project was to build a classifier that could help doctors detecting donors with the metabolic syndrome.

In a Bayesian framework, for each donor we therefore computed the posterior probability $p_i$ that each of the estimated target variables was in the critical range, producing an alert if at least 3 targets out of 5 were at risk. This would clinically serve as a need for the doctor to make further investigations, basically measuring the target variables only for the critical donors.

In our problem, the misclassification costs were clearly not equal, because letting an ill donor donate blood (i.e, a false negative) is much more dangerous than denying the donation of a healthy donor (i.e, a false positive). Therefore, the "alert threshold" (that is, the maximum posterior probability $p^*$ such that if $p_i > p^*$, the donor would be classified as "at risk") was computed in order to minimize the number of false negatives.

Table 12 shows the confusion matrix in the case where no error on false negatives has to be made, while Table 13 allows for a 5% error rate.

| Actual \ Predicted | YES | NO |
|---|---|---|
| YES | 241 | 0 |
| NO | 1005 | 2928 |

Table 12: Confusion matrix for 0% false negatives

| Actual \ Predicted | YES | NO |
|---|---|---|
| YES | 229 | 12 |
| NO | 517 | 3416 |

Table 13: Confusion matrix for 5% false negatives

The corresponding thresholds were 0.009406416 for 0% false negatives and 0.07189475 for 5% false negatives. The fact that both these thresholds are not incredibly high is coherent with the fact that the metabolic syndrome is a rare disease, and therefore even a "small" probability is enough to raise an alert.

## 7. Conclusion

In our project we addressed the challenge of predicting metabolic syndrome in blood donors using longitudinal data. Despite the presence of substantial missing values in the target variables, we managed to simulate the missing data, effectively recovering valuable observations.

Our investigation into autoregressive components and interdependencies between the targets enhanced the model's ability to capture temporal dynamics and relationships within the data.

By iteratively incorporating predictions from previous targets as covariates within Bayesian linear models, we achieved an improvement in predictive performance, as evidenced by an enhancement in the Bayesian $R^2$. Moreover, our results converged consistently, highlighting the robustness of the procedure.

Finally, the development of a classifier to estimate the probability of metabolic syndrome, with a focus on minimizing false negatives, ensured the clinical reliability of the model.

Overall, our work demonstrates a solid application of statistical methods to longitudinal data, providing a promising tool for predicting metabolic syndrome, with important implications for health monitoring in blood donors.

## 8. Further Developments

1. **Beyond the donor-specific intercept:** Currently, the only donor-specific term in the model is the intercept. However, a further step can be made. By allowing the regression coefficients $\beta$ to vary individually for each donor, we could build an even more personalized regression model for each donor. Therefore, each donor's risk of developing metabolic syndrome could be predicted more accurately.
2. **Clustering donors to make the model more general:** One limitation of the current model is its lack of generalizability to new donors. The presence of a donor-specific random intercept means that the model cannot be applied directly to new donors without being refit. To address this, a clustering approach could be implemented to group donors with similar characteristics. With this approach, the model would become far more general. Moreover, if combined with point 1, the model would also have fewer parameters than before (the $\beta$ coefficients would now be cluster-specific instead of donor-specific). This would also reduce the risk of overfitting, which is a problem especially in the case where few data for each donor / cluster are available.

## References


[1] A. Scuteri et al. "Metabolic syndrome across Europe: Different clusters of risk factors". In: *European Journal of Preventive Cardiology* 22.4 (2015), pp. 486–491.

[2] S. M. Mohamed et al. "Metabolic syndrome: Risk factors, diagnosis, pathogenesis, and management with natural approaches". In: *Food Chemistry Advance* 3 (2023).

[3] Francesca Arrigoni. "Bayesian models for early diagnosis and prediction of metabolic syndrome in healthy blood donors". MA thesis. Politecnico di Milano, 2023-2024.

[4] Amos Tirosh et al. "Changes in triglyceride levels over time and risk of type 2 diabetes in young men". In: *Diabetes Care* 31.10 (2008), pp. 2032–2037.

[5] Ping-Yu Chang et al. "Triglyceride Levels and Fracture Risk in Midlife Women: Study of Women's Health Across the Nation (SWAN)". In: *The Journal of Clinical Endocrinology & Metabolism* 101.9 (2016), pp. 3297–3305.

[6] Alan M. Garber et al. "Predicting High-Risk Cholesterol Levels". In: *International Statistical Review* 62.2 (1994), pp. 203–228.

[7] Johan G Eriksson Reijo Siren and Hannu Vanhanen. "Waist circumference a good indicator of future risk for type 2 diabetes and cardiovascular disease". In: *BMC Public Health* 12.631 (2012).

[8] J. Liu et al. "Long-term glucose forecasting using a physiological model and deconvolution of the continuous glucose monitoring signal". In: *Sensors* 19 (2019).

[9] Gianfranco Parati et al. "Spectral Analysis of Blood Pressure and Heart Rate Variability in Evaluating Cardiovascular Regulation: A Critical Appraisal". In: *Hypertension* 25.6 (1995), pp. 1276–1286.

[10] R. Mohammadi and E.C. Wit. "BDgraph: An R package for Bayesian Structure Learning in Graphical Models". In: *Journal of Statistical Software* 89.3 (2019), pp. 1–30.

[11] A. Gelman et al. "R-squared for Bayesian Regression Models". In: *The American Statistician* 73.3 (2019), pp. 307–309.

# A. Stan code

## A.1. MCMC imputation

```
data{
    int<lower=1> N; //number of subjects
    int<lower=1> P; // number of covariates
    int<lower=1> T; // number of total observations
    int<lower=1> subj[T]; // subject id vector

    int<lower=0> n_obs; // number of observed y
    int<lower=0> n_miss; // number of missing y
    int<lower=1, upper=T> obs_indices[n_obs]; // indices of observed y
    int<lower=1, upper=T> miss_indices[n_miss]; // indices of missing y
    vector[n_obs] y_obs;// observed y values

    matrix[n_obs, P] X1;
    matrix[n_miss, P] X2;
}
parameters {
    vector[P] beta; // fixed intercept and slope
    vector[N] b; // subject intercepts
    real<lower=0> eta; // sd for subject intercepts
    real<lower=0> sigma_e; // error sd
    real mub; // mean for subject intercepts
}
model {
// Priors
    mub ~ normal(0, 2); // subj random effects
    eta ~ inv_gamma(3, 2); // variance of intercepts
    b ~ normal(mub, eta); // subj random effects
    beta ~ normal(0, 5); // fixed effects
    sigma_e ~ inv_gamma(3, 2);// error variance

// Linear predictor for observed y (just estimate mu on known target values)
    vector[n_obs] mu_obs;
    mu_obs = X1 * beta + b[subj[obs_indices]];

// Likelihood for observed y
    y_obs ~ normal(mu_obs, sigma_e);

}
generated quantities {
    vector[n_obs] y_obs_hat; // Predictions for observed y
    vector[n_miss] y_miss_hat; // Predictions for missing y
    vector[T] y_hat; // Combined predicted y for all observations

// Fill in predictions for observed y
    y_obs_hat = X1 * beta + b[subj[obs_indices]]; // computes y_obs_hat from linear model
    y_hat[obs_indices] = y_obs_hat;

// Compute mu_miss for missing y
    vector[n_miss] mu_miss;
    mu_miss = X2 * beta + b[subj[miss_indices]];

// Since we want to simulate data, we sample y_miss_hat
// from a normal distribution with mean mu_miss and variance sigma_e
    for (i in 1:n_miss) {
       y_miss_hat[i] = normal_rng(mu_miss[i], sigma_e);
    }
```

```
// Fill in y_hat for missing indices
    y_hat[miss_indices] = y_miss_hat;
}
```

## A.2. Autoregressive model

```
data {
    int<lower=1> N; //number of subjects
    int<lower=1> P; //number of covariates
    int<lower=1> T; //number of total observations = dates(first donor) + dates(second) + ...
    int<lower=1> subj[T]; //subject id vector
    real y[T]; //outcome
    vector[T] y_trasl; //shifted outcome for autoregression
    matrix[T,P] X; //predictors
}
parameters {
    vector[P] beta; //fixed intercept and slope
    vector[N] b; //subject intercepts (note:  fixed for each donor, not time-dependent)
    real<lower=0> eta; //sd for subject intercepts
    real<lower=0> sigma_e; //error sd
    real mub;
    real phi; //autoregressive coefficient
}
// The model to be estimated.  We model the output
// 'y' to be normally distributed with mean 'mu'
// and standard deviation 'sigma'.

model {
    //priors
    mub ~ normal(0, 2); //subj random effects
    eta ~ inv_gamma(3,2); //variance of intercepts
    b ~ normal(mub, eta); //subj random effects
    beta ~ normal(0,5);
    sigma_e ~ inv_gamma(3,2);
    phi ~ normal(0, 10); //if you don't want it to be a uniform

    vector[T] mu;

    mu = (X * beta + b[subj]) + (phi * y_trasl);

    y ~ normal(mu, sigma_e);
}
generated quantities {
    vector[T] y_hat;

    y_hat = (X * beta + b[subj]) + (phi * y_trasl);
}
```

## A.3. Final model: model for *Circonferenza vita*

```
data {
    int<lower=1> N; // number of subjects
    int<lower=1> P; // number of covariates
    int<lower=1> T; // number of total observations
    int<lower=1> subj[T]; // subject id vector train
    vector[T] y; // train y values
    matrix[T, P] X; // predictors train values y
```

```
    matrix[N, P] Xtest; // predictors test values y
    int<lower=1> subj_test[N]; // subject id vector test
}
parameters {
    vector[P] beta; // fixed intercept and slope
    vector[N] b; // subject intercepts
    real<lower=0> eta; // sd for subject intercepts
    real<lower=0> sigma_e; // error sd
    real mub; // mean for subject intercepts
}
model {
    // Priors
    mub ~ normal(0, 2); // subj random effects
    eta ~ inv_gamma(3,2); // variance of intercepts
    b ~ normal(mub, eta); // subj random effects
    beta ~ normal(0, 5); // fixed effects
    sigma_e ~ inv_gamma(3, 2); // error variance

    // Linear predictor for observed y (just estimate mu on known target values)
    vector[T] mu = X * beta + b[subj];

    // Likelihood for test y
    y ~ normal(mu, sigma_e);

}
generated quantities
    vector[T] y_train; // Predictions for train y
    vector[N] y_test; // Predictions for test y

    //y train predictions
    y_train = X * beta + b[subj];

    //y test predictions
    y_test = Xtest * beta + b[subj_test];
}
```

## A.4. Final model: model for all the other target variables

```
data {
    int<lower=1> N; // number of subjects
    int<lower=1> P; // number of covariates
    int<lower=1> T; // number of total observations
    int<lower=1> K; // previous targets
    int<lower=1> subj[T]; // subject id vector train
    vector[T] y; // train y values
    matrix[T, P] X; // predictors train values y
    int<lower=0> lag; // threshold for autocorrelation
    matrix[T,K] y_std; // previous targets standardized
    matrix[T,K] y_pred_train_std; // previous predicted targets standardized
    vector[T] y_trasl; // autoregression values

    matrix[N, P] Xtest; // predictors test values y
    int<lower=1> subj_test[N]; // subject id vector test
    matrix[N,K] y_pred_test_std; // previous predicted targets standardized
    vector[N] y_trasl_test; // autoregression values
}

parameters {
    vector[P] beta; // fixed intercept and slope
```

```
    vector[N] b; // subject intercepts
    vector[K] gamma; // targets parameters
    real<lower=0> eta; // sd for subject intercepts
    real<lower=0> sigma_e; // error sd
    real mub; // mean for subject intercepts
    real phi_raw; //autoregressive coefficient


transformed parameters {
    real phi;
    //if no autoregressive component set it to zero
    if(lag!=0){
       phi = phi_raw;
    }
    else {
       phi = 0;
    }
}

model {
    // Priors
    mub ~ normal(0, 2); // subj random effects
    eta ~ inv_gamma(3, 2); // variance of intercepts
    b ~ normal(mub, eta); // subj random effects
    beta ~ normal(0, 5); // fixed effects
    sigma_e ~ inv_gamma(3, 2); // error variance
    gamma ~ normal(0, 5); // previous targets effects
    phi_raw ~ normal(0, 1); // from previous experiments we know this value is really low
    // Linear predictor for observed y (just estimate mu on known target values)
    vector[T] mu = X * beta + b[subj] + (phi * y_trasl) + y_std * gamma;

    //values associated to autoregressive component NOT standardized since output is log-transformed
    // and here we are computing autoregressive value

    // Likelihood for test y
    y ~ normal(mu, sigma_e);

}
generated quantities {
    vector[T] y_train; // Predictions for train y
    vector[N] y_test; // Predictions for test y

    //y train predictions
    y_train = X * beta + b[subj] + (phi * y_trasl) + y_pred_train_std * gamma;
    //y test predictions
    y_test = Xtest * beta + b[subj_test] + (phi * y_trasl_test) + y_pred_test_std * gamma;
}
```