Supplement 1 - modgo demonstration
A simple-to-use R package for mimicking study data by simulations

By Francisco Ojeda, George Koliopanos, and Andreas Ziegler

Table of contents

1 Data set

For illustration, we selected the Cleveland Clinic Heart Disease Data set from the University of California in Irvine (UCI) machine learning data repository (Dua and Graff 2017). Below, we are using eleven variables, five of which are continuous, four are dichotomous, and two ordinal categorical variables.

```
# Specifying dichotmous and ordinal categorical variables
binary_variables <- c("Sex","HighFastBloodSugar","CAD","ExInducedAngina")
categorical_variables <- c("Chestpaintype","RestingECG")

nrep <- 500
```

2 Default modgo run

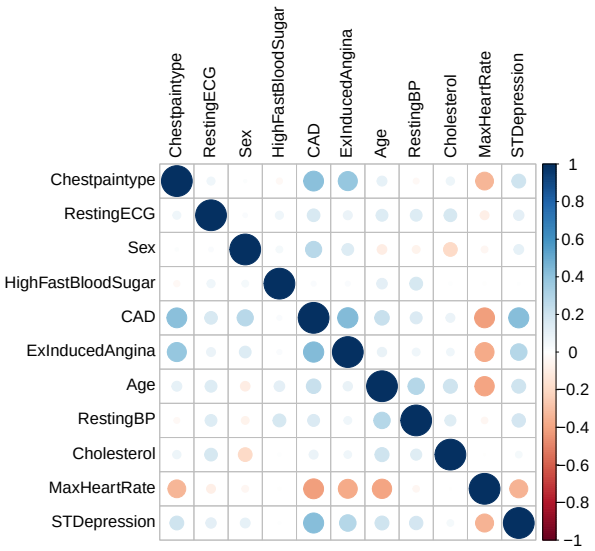
In this section, we run *modgo* with his default settings. For *modgo* to produce results that mimic the original data set efficiently, user needs to specify dichotomous and ordinal categorical variables. Variables will be considered as continuous, otherwise. All *modgo* runs in this and the following sections will produce 500 data sets with the specificition nrep = 500; the default is 100.

Figure 1 shows the correlation plots for the default *modgo* run, and Figure 2 displays the distribution plots for the original data set and one simulated data set. The default displayed simulated data set is the first one.

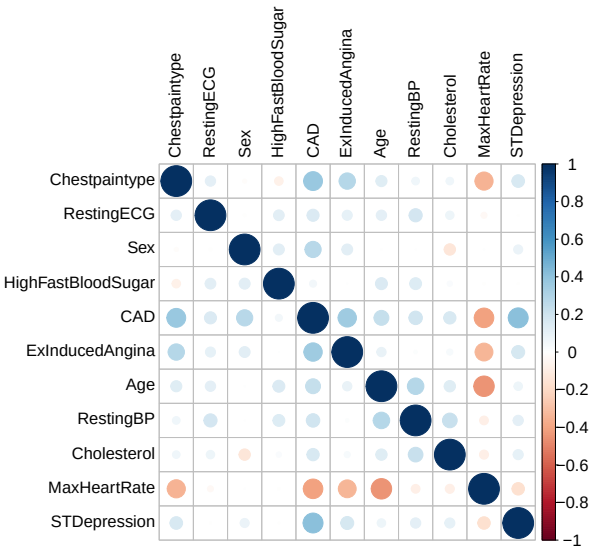
```
test <- modgo(data = dataset,bin_variables = binary_variables,
              categ_variables = categorical_variables,nrep = nrep)
```

Figure 1: Correlation plots for a default *modgo* run. This figure shows the original correlations, the correlations from a single simulated data set – default is the first simulated data set –, the mean correlation matrix of all simulated data sets, and the difference between the two correlation matrices from the original data and the single simulated data set.

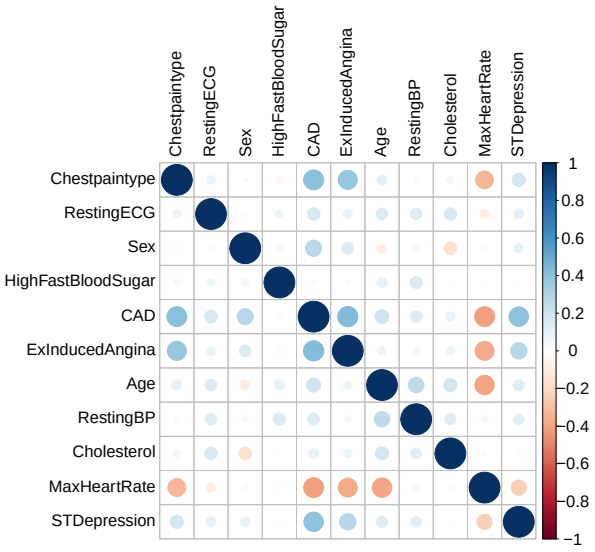
Original



Simulated



Mean correlation of simulations



Original minus simulated

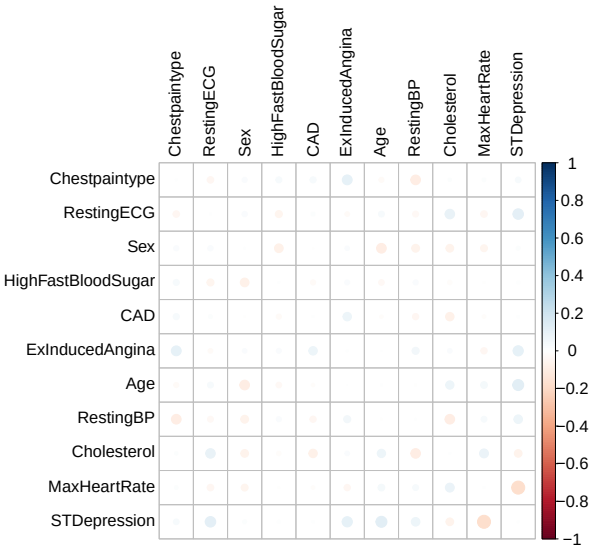
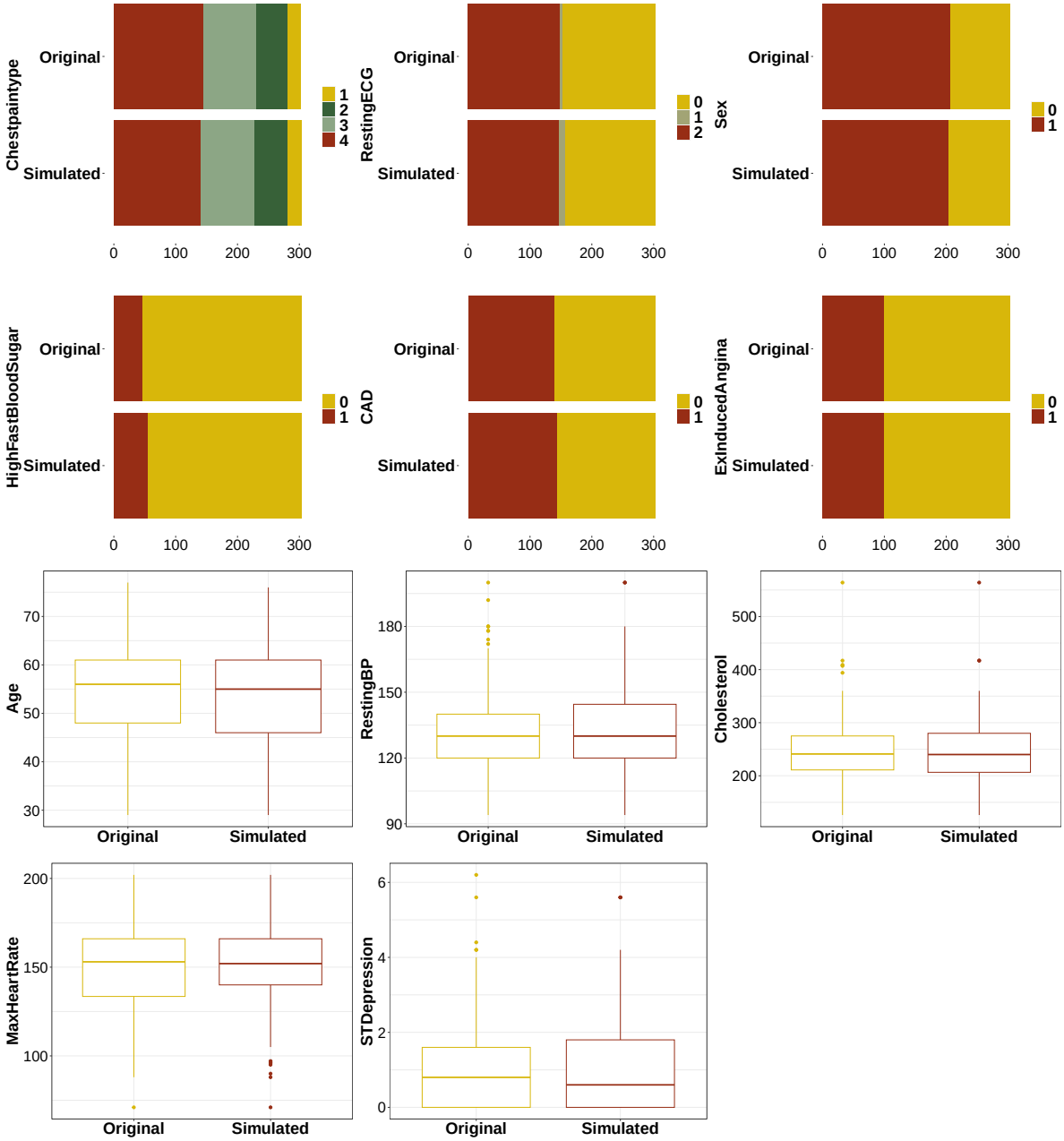


Figure 2: Distribution plot depicts distribution for each variable of the original data set and of a simulated data set (default: 1st simulated data set). Box plots are used for continuous variables. Bar plots are used for binaries and ordinal categorical variables.



3 Expansion 1: Selection by thresholds of variables

modgo provides an option so that only subjects (instances) are simulated that fulfill a specific requirement. In the simplest case (Section 3.1), the user can specify an upper or a lower boundary, or an interval for a variable. The use may alternatively specify a combination of variables and thresholds (Section 3.2).

3.1 Threshold for a single variable

Three steps are required when subjects need to fulfill a specific selection criterion for a continuous variable. First, the name of the variable needs to be specified, for which the threshold needs to be set. Second, the left and right boundaries need to be specified. Third, a dataframe with three columns is defined with Column 1: variable name of threshold variable, Column 2: left boundary, i.e., lower bound, Column 3: right boundary, i.e., upper bound. Finally, the dataframe is imported using the *thresh_var* argument. In the example, all subjects have to be at least 66 years old. The selection variable therefore is *Age* with left threshold *65* and right threshold infinity *NA*.

If the percentage of samples fulfilling the indicated threshold requirements are less than 10% of the simulated samples, *modgo* stops to avoid excessive computation time. However, users can force *thresh_force* = *TRUE* the requested simulation to be run.

Figure 3 shows the correlation plot for this illustration. Substantial differences between the original and the simulated correlation plots can be observed for the maximum heart rate and several other variables. Figure 4 displays the corresponding distribution plot. The age distribution is shifted as expected. Furthermore, the distribution of subjects with coronary artery disease (CAD = 1) is higher in the simulated than the original data set.

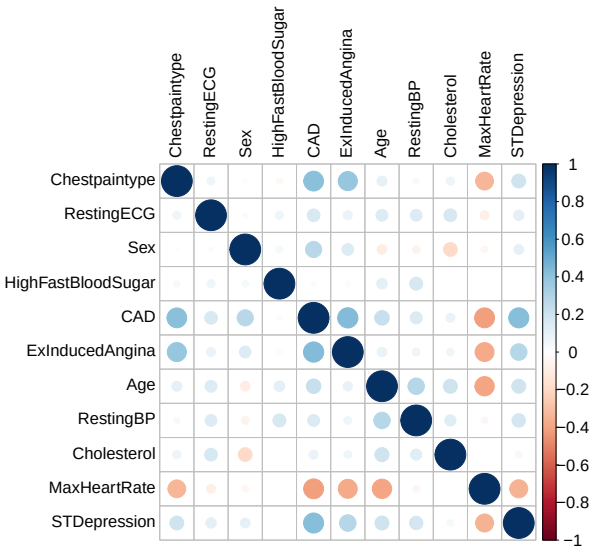
```
Variables <- c("Age")
thresh_left <- c(65)
thresh_right <- c(NA)
thresholds <- data.frame(Variables, thresh_left, thresh_right)

print(as.matrix(thresholds))
```

```
     Variables thresh_left thresh_right
[1,] "Age"     "65"        NA
```

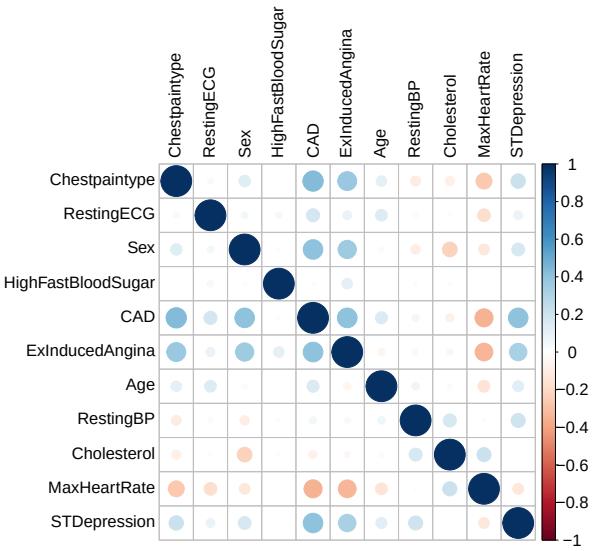
```
test_1 <-  modgo(data = dataset,bin_variables = binary_variables,
                 categ_variables = categorical_variables,
                 thresh_var = thresholds,nrep = nrep)
```

Figure 3: Correlation plots for a *modgo* run to only include subjects with age of at least 66 years, i.e., a run that uses a threshold argument. Displayed are the correlations of the original data set, the correlations of a single simulated data set which fulfills the selection criterion according to the specified age threshold, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

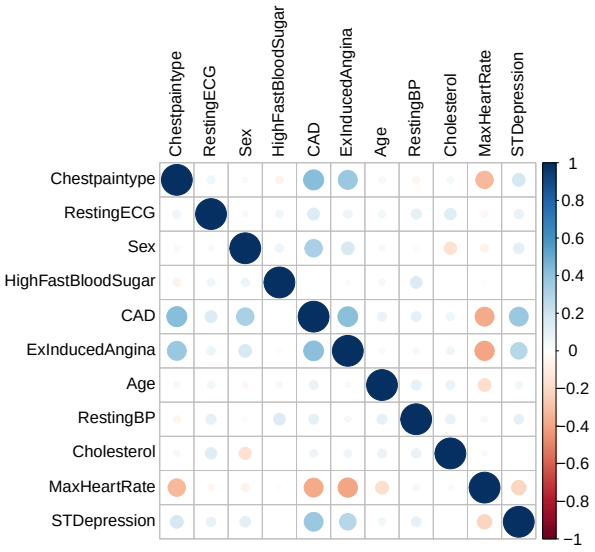
Original



Simulated



Mean correlation of simulations



Original minus simulated

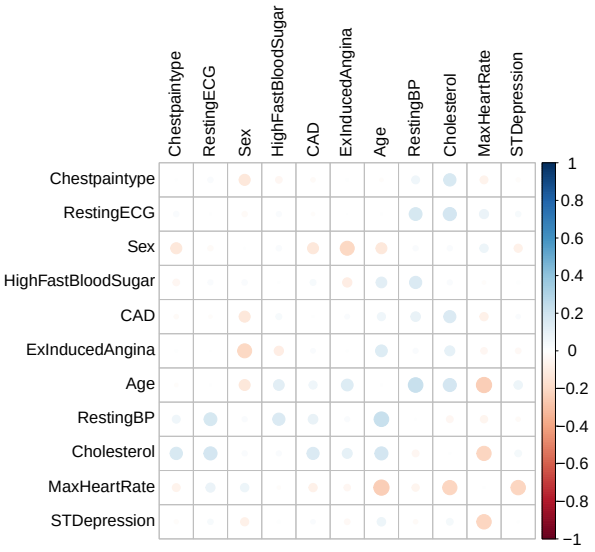
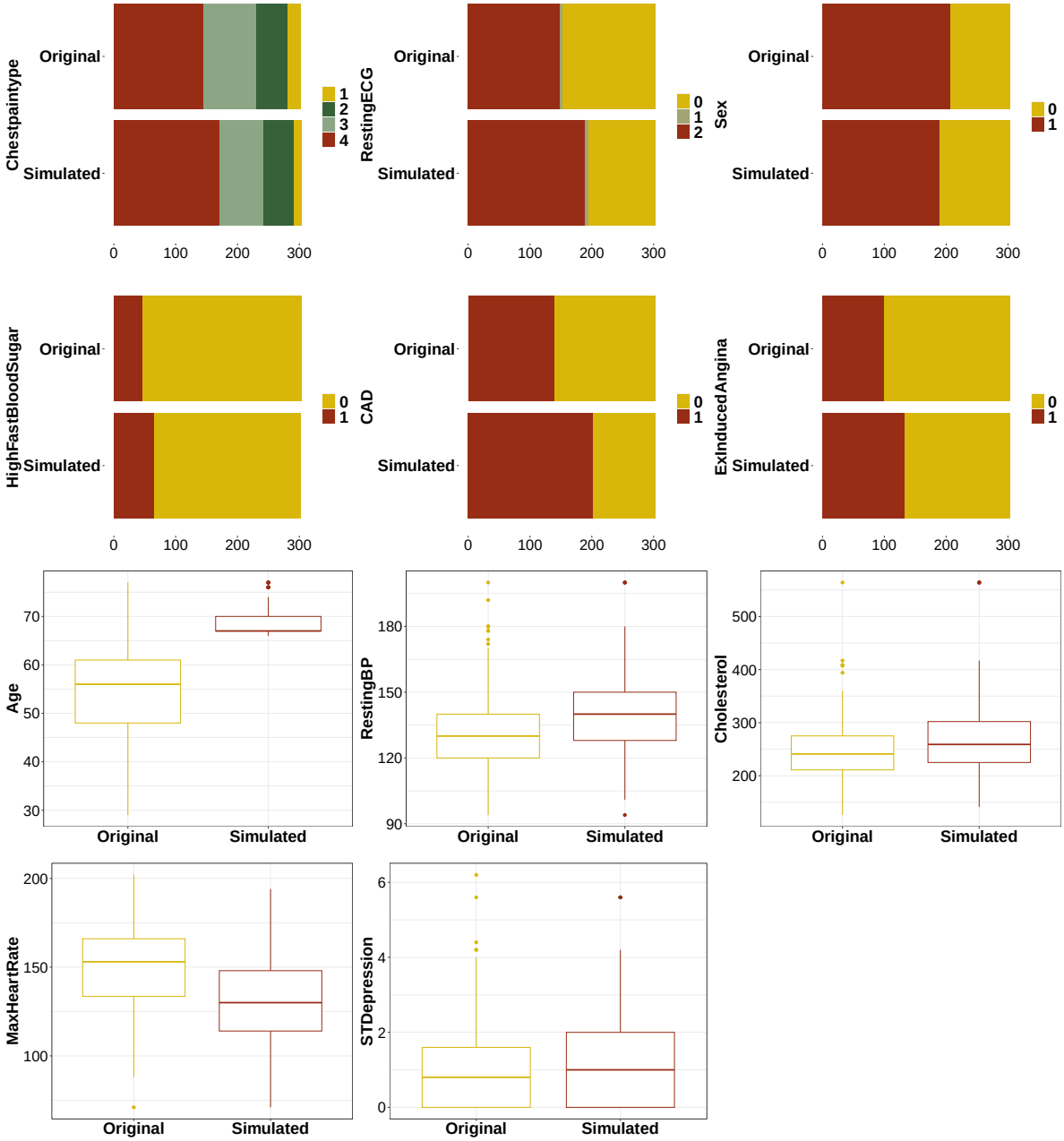


Figure 4: Distribution plot to depict the distribution for each variable of the original data set and a single simulated data set. In the simulations, only subjects with age of at least 66 years were included.



3.2 Thresholds for multiple variables

In this section, we select subjects by specifying thresholds for three variables, specifically age > 65 years, max heart rate < 200 beats per minute (bpm) and resting blood pressure between 100 mmHg and 150 mmHg.

Analogously to the previous examples, Figure 5 and Figure 6 show the correlation plots and distribution plots, respectively.

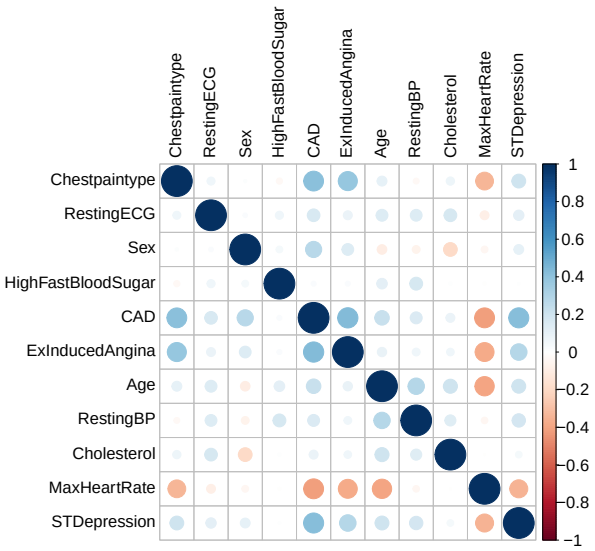
```
Variables <- c ("Age","MaxHeartRate","RestingBP")
thresh_left <- c(65,NA,100)
thresh_right <- c(NA,200,150)
thresholds <- data.frame(Variables, thresh_left, thresh_right)

print(as.matrix(thresholds))
```

```
     Variables      thresh_left thresh_right
[1,] "Age"          " 65"       NA
[2,] "MaxHeartRate" NA          "200"
[3,] "RestingBP"    "100"       "150"
```

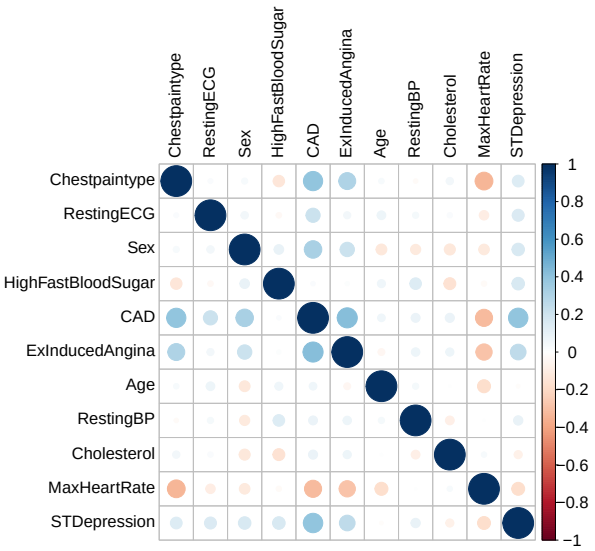
```
test_2 <-  modgo(data = dataset,bin_variables = binary_variables,
                 categ_variables = categorical_variables,
                 thresh_var = thresholds,thresh_force = TRUE,nrep = nrep)
```

Figure 5: Correlation plots for a *modgo* run that used the threshold argument for the three variables age, maximum heart rate, and resting blood pressure. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

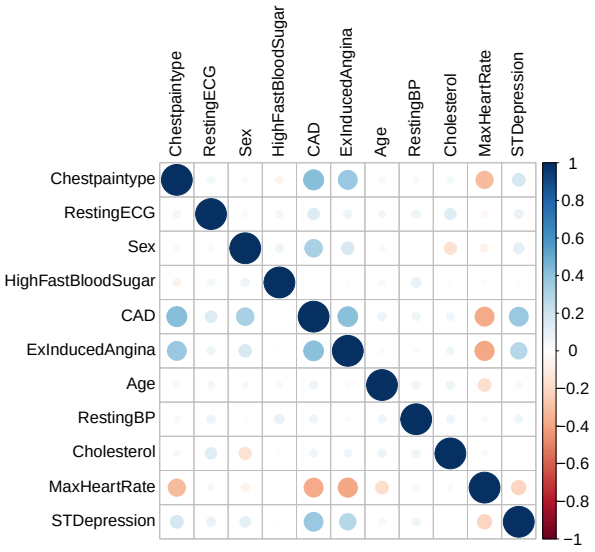
Original



Simulated



Mean correlation of simulations



Original minus simulated

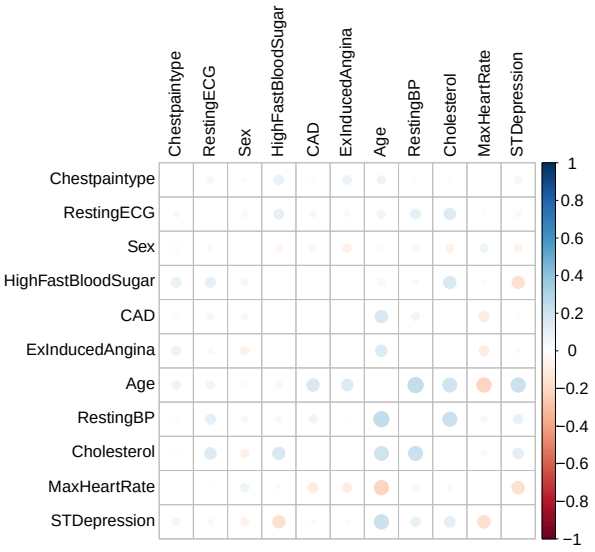
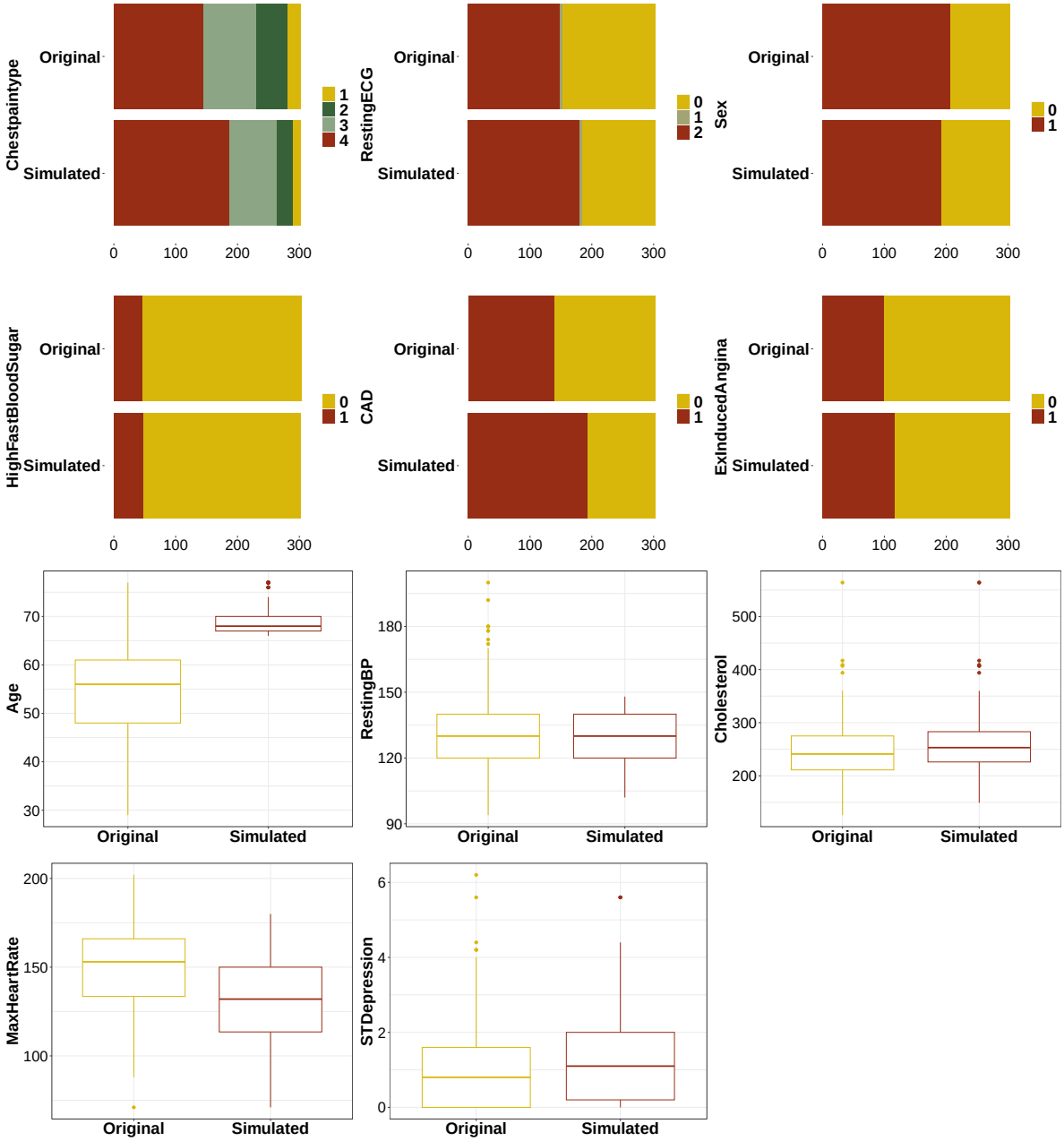


Figure 6: Distribution plots depicting the distribution for each variable of the original data set and a single simulated data set. For the simulations, thresholds were set for the three variables age, maximum heart rate, and resting blood pressure.



4 Expansion 2: Perturbation analysis

Perturbations may be used to increase the difference between the original data set and simulated data sets. We introduce two approaches for perturbation analysis through *modgo*. The first approach keeps the variance of the perturbed variable unchanged (Section 4.1), while the second intend to increase it to a specified amount (Section 4.2).

4.1 Perturbation analysis - Unchanged variance

For continuous variables, *modgo* provides the option to add a normally distributed noise with mean 0 and variance σ_p^2. With this perturbation, the variance of the perturbed variable is identical to the variance of the original variable. This option permits the generation of values from continuous variables, which were not observed in the original data set.

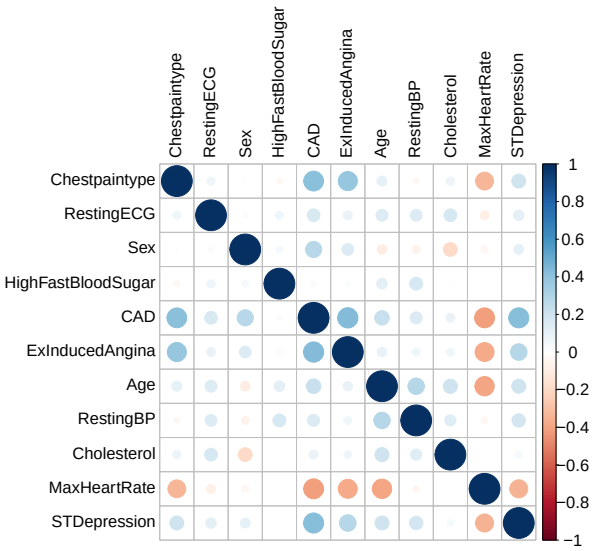
To specify which variables are to be perturbed and to which degree, i.e., percentage, the user needs to provide *modgo* with a named vector of the percentages and with the corresponding variables names as the names of the vector.

Similar to the previous examples, Figure 7 shows the correlation plots for the expansion to perturbations, and Figure 8 displays the distribution plots. Figure 8 shows that the distribution of both resting blood pressure and cholesterol change substantially due to the perturbation.

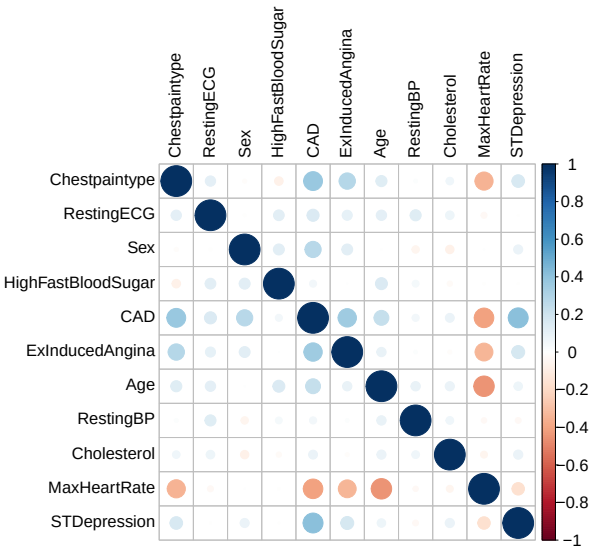
```
#Create named vector
perturb_vector <- c(0.9,0.7)
names(perturb_vector) <- c("RestingBP","Cholesterol")

test_3 <-  modgo(data = dataset,bin_variables = binary_variables,
                 categ_variables = categorical_variables,
                 pertr_vec = perturb_vector,nrep = nrep)
```

Figure 7: Correlation plots for a modgo run using the perturbation extension. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

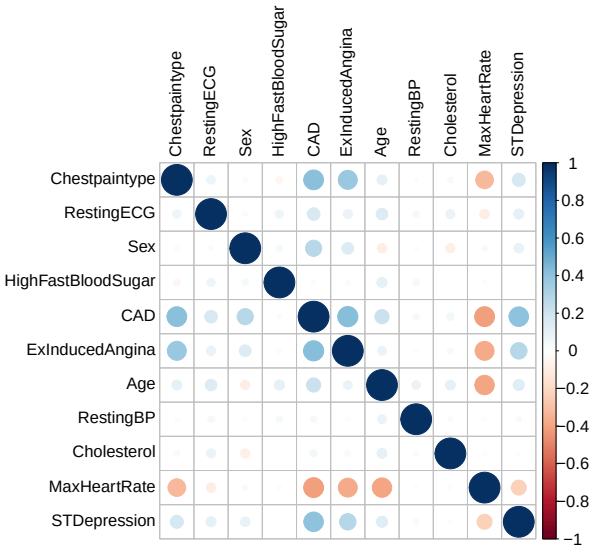
Original



Simulated



Mean correlation of simulations



Original minus simulated

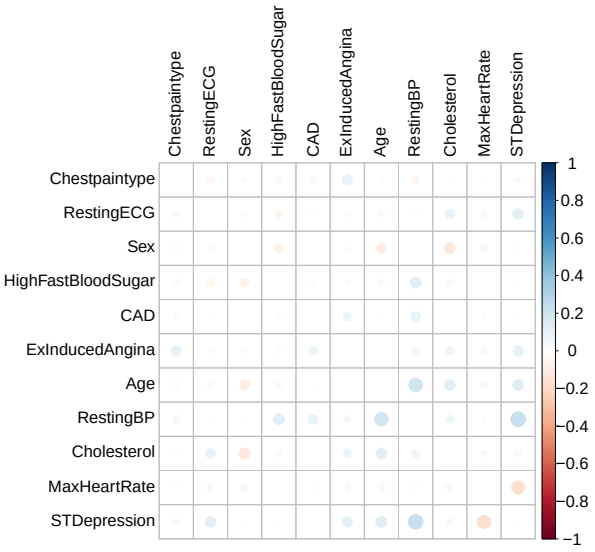
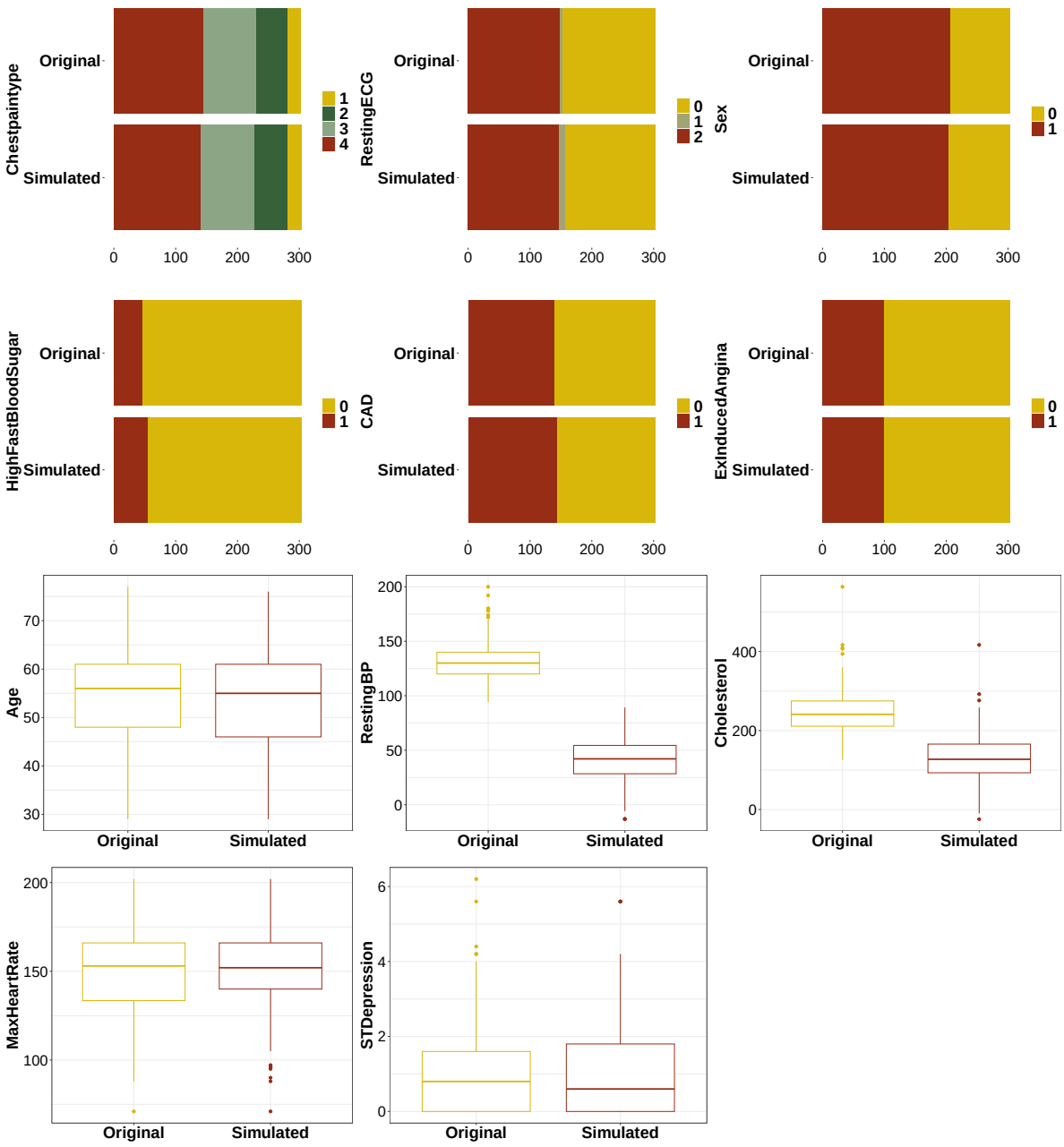


Figure 8: Distribution plot to depict the distribution for each variable of the original data set and a single simulated data set with perturbed resting blood pressure and cholesterol levels.



4.2 Perturbation analysis - Increase continuous variables variances

One more option that modgo provides is adding a normally distributed noise with mean 0 and variance $\sigma_p^2 = p * \sigma^2$($\sigma^2$ = original data set´s variance). With this perturbation, the variance of the perturbed variable is p times bigger than the variance of the original variable.

To specify which variables are to be perturbed and to which degree you want to increase their variances, the user needs to provide *modgo* with a named vector of the multipliers and with the corresponding variables names as the names of the vector. For this example, we request that resting blood pressure simulation has 2 times bigger variance than the original data, and cholesterol 20 times.

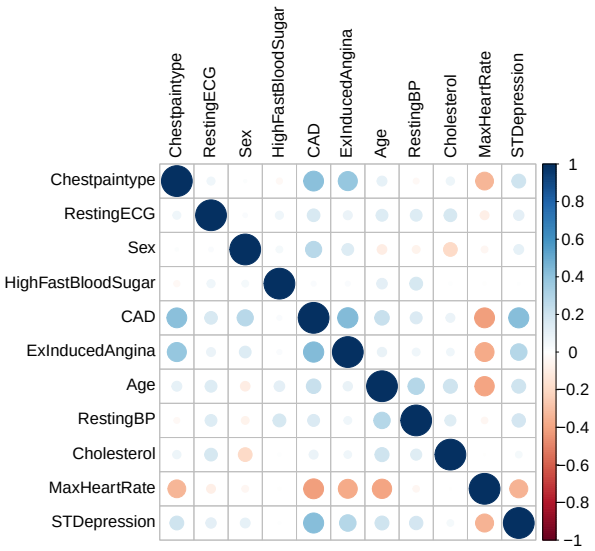
By default, *modgo* perturbation with increased variance is applied to the simulated data set. This may produce simulated data sets with different correlation matrices from the original correlation. For this reason, *modgo* offers the option to perform this particular perturbation to the original data before simulation begins, while using unchanged original data correlation matrix as the covariance matrix. With this method, correlations do not change significantly, while the variance of the variable is increased to the requested amount. To enable this option, the user needs to include *infl_cov_stable = TRUE*.

In Figure 9 and 10, we illustrate the correlation and distribution plots for the perturbation analysis with increased variance, while *infl_cov_stable = FALSE*. Figure 11 and 12 depict correlation and distribution plots for the perturbation analysis with increased variance, using *infl_cov_stable = TRUE* argument. By looking at the correlation plots, we can observe that Cholesterol loses its correlations when we use *infl_cov_stable = FALSE*.

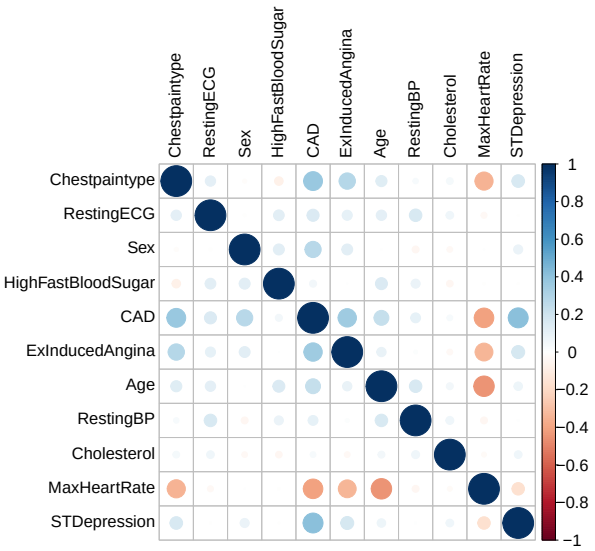
```
#Create named vector
var_inflation_vector <- c(2,20)
names(var_inflation_vector) <- c("RestingBP","Cholesterol")

test_var_infl_1 <-  modgo(data = dataset,bin_variables = binary_variables,
                  categ_variables = categorical_variables,
                  var_infl = var_inflation_vector,nrep = nrep,infl_cov_stable = FALSE)
```

Figure 9: Correlation plots for a *modgo* run using the perturbation extension with increased variance. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

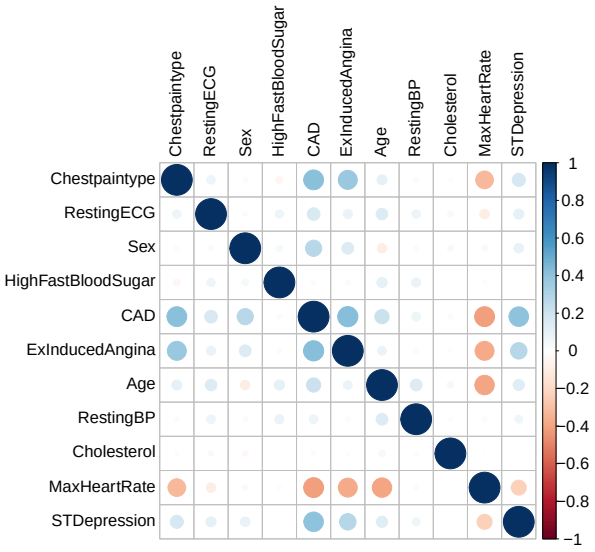
Original



Simulated



Mean correlation of simulations



Original minus simulated

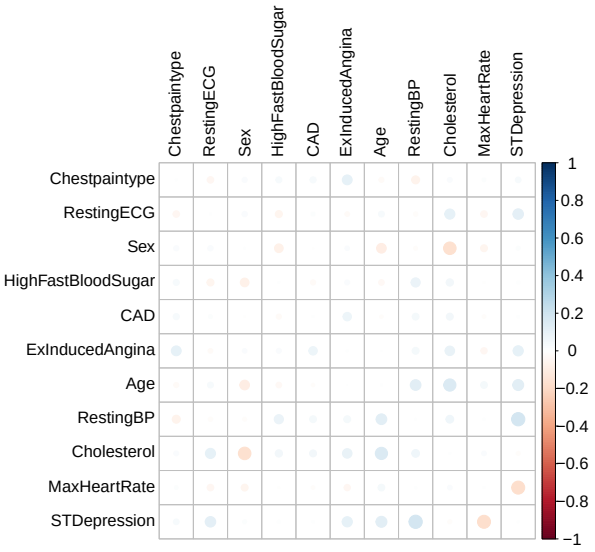
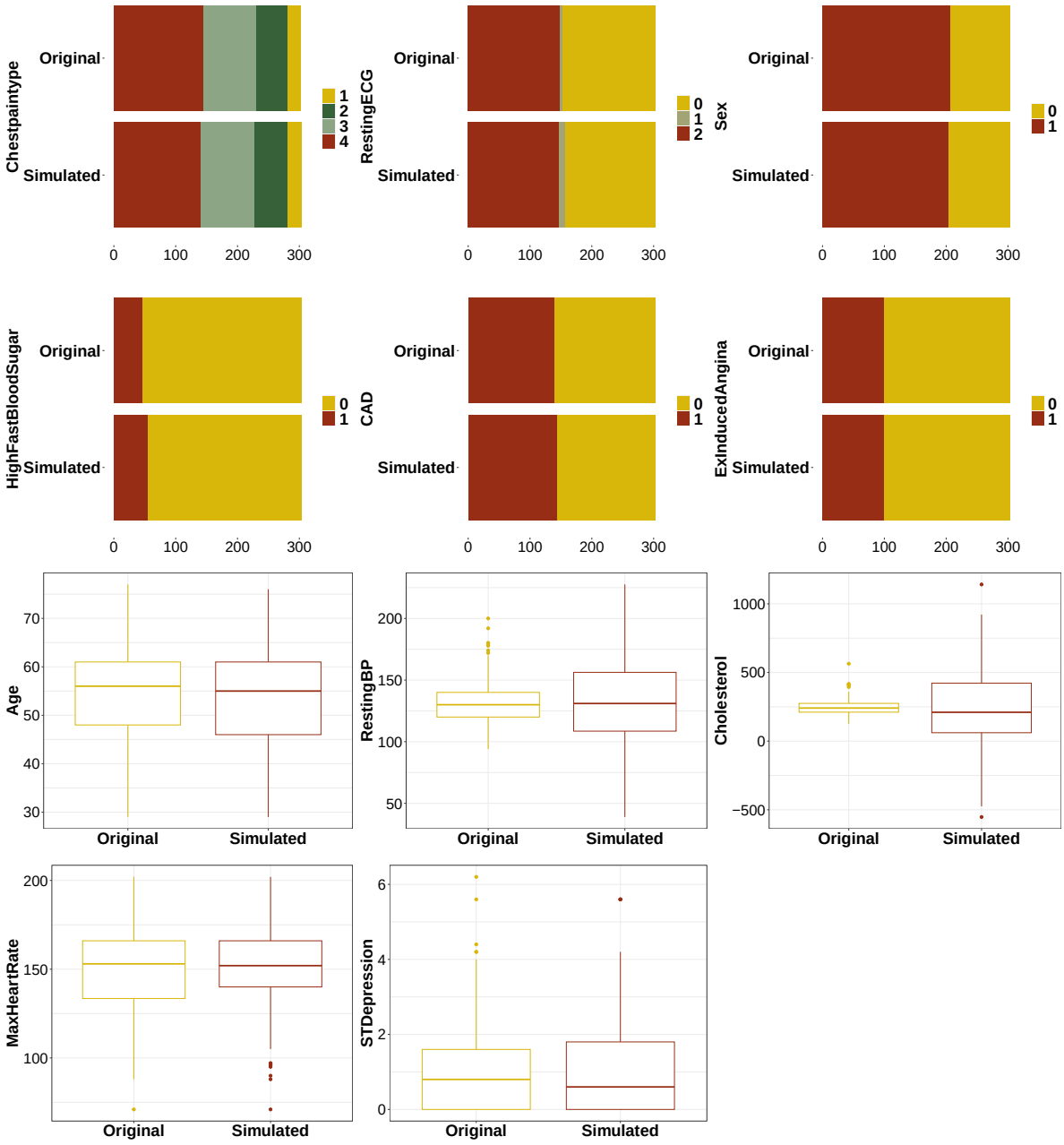


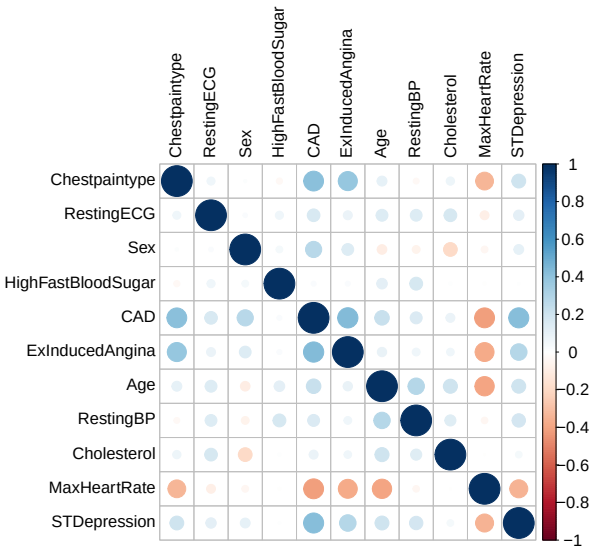
Figure 10: Distribution plot to depict the distribution for each variable of the original data set and a single simulated data set with perturbed resting blood pressure and cholesterol levels.



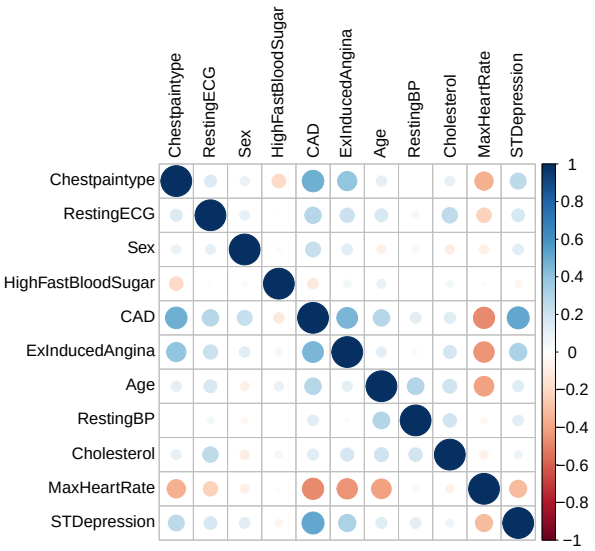
```r
#Create named vector

test_var_infl_2 <-  modgo(data = dataset,bin_variables = binary_variables,
                 categ_variables = categorical_variables,
                 var_infl = var_inflation_vector,nrep = nrep,infl_cov_stable = TRUE)
```

Figure 11: Correlation plots for a *modgo* run using the perturbation extension with increased variance without chaning significantly the correlations. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

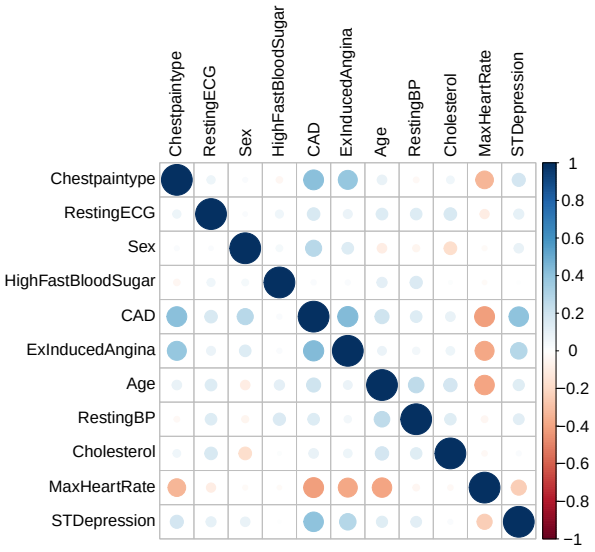
Original



Simulated



Mean correlation of simulations



Original minus simulated

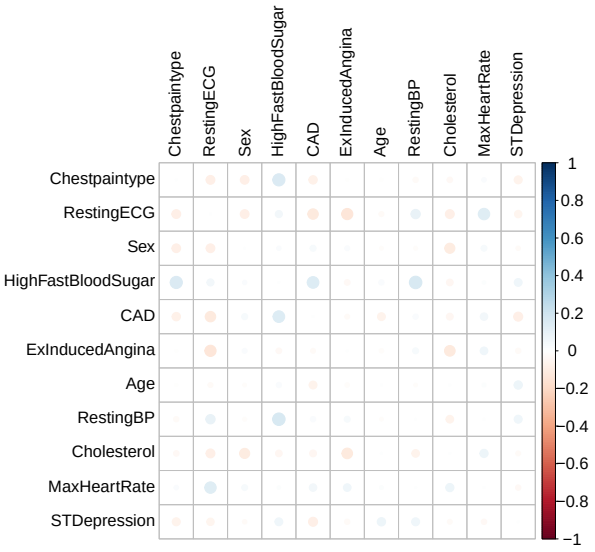
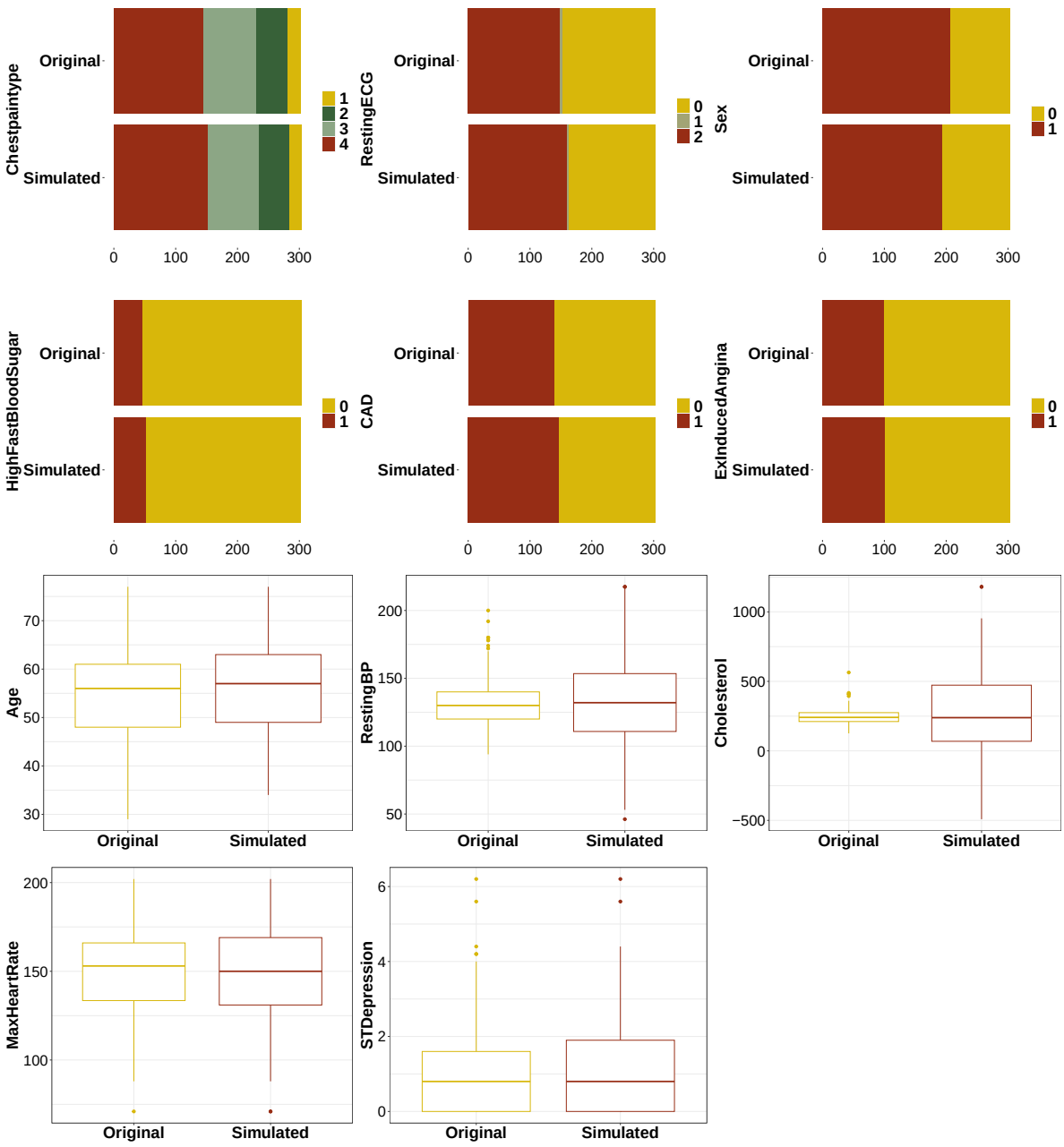


Figure 12: Distribution plot to depict the distribution for each variable of the original data set and a single simulated data set with perturbed resting blood pressure and cholesterol levels.



5 Expansion 3: Proportion of dichotomous variables

A user may request *modgo* to simulate data sets with a specific proportion of case-control samples. *modgo* will then produce data sets that have the requested proportion or at least the specified proportion. Specifically, if half of 7 are to have a certain disease, *modgo* will produce data sets with 4 diseased samples out of 7, which is 57%, i.e., the ceiling function will be used internally.

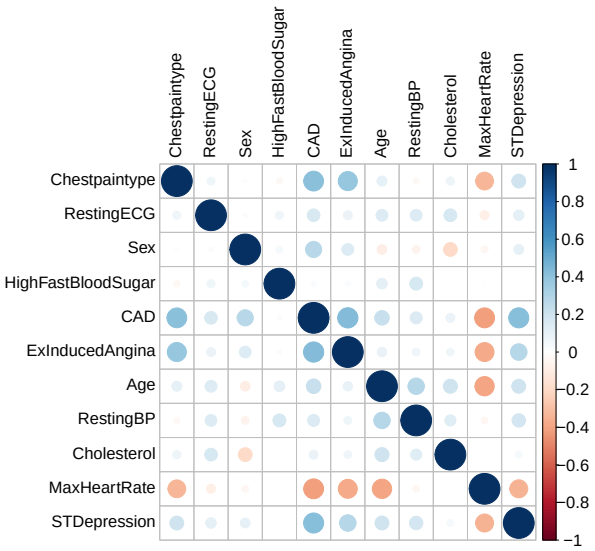
In the illustrative code below, 90% of the samples in each simulated data set were requested to have coronary artery disease. To this end, a named vector with the name of the dichotomous variable needs to be defined together with the requested proportion.

Figures 13 and 14 display the resulting correlation plots and distribution plots, respectively.

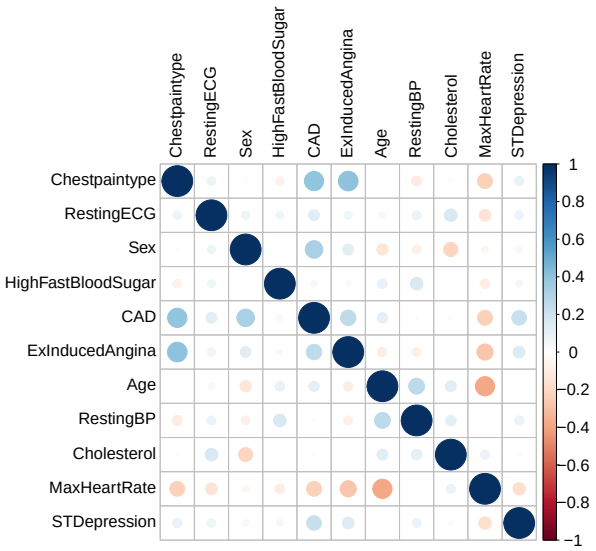
```
#Create named vector
CAD_proportions <- c(0.9)
names(CAD_proportions) <- c("CAD")

test_4 <-  modgo(data = dataset,bin_variables = binary_variables,
                 categ_variables = categorical_variables,
                 var_prop = CAD_proportions,nrep = nrep)
```

Figure 13: Correlation plots for a *modgo* run with altered proportions of the dichotomous variable coronary artery disease. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

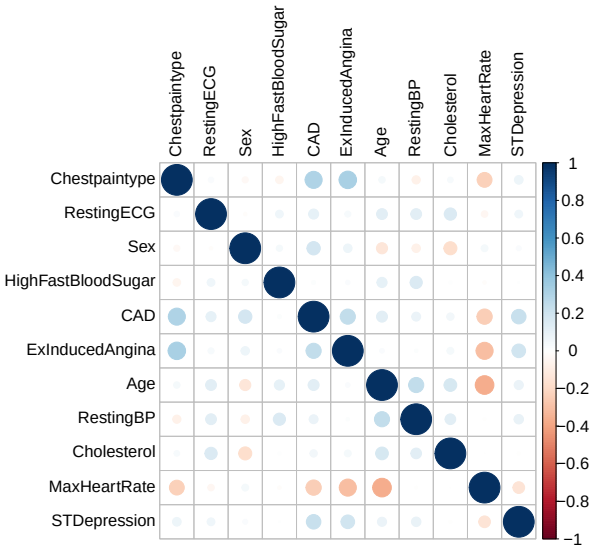
Original



Simulated



Mean correlation of simulations



Original minus simulated

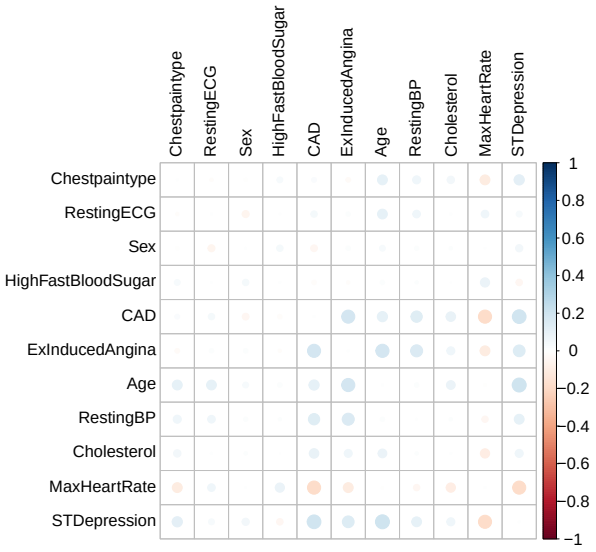
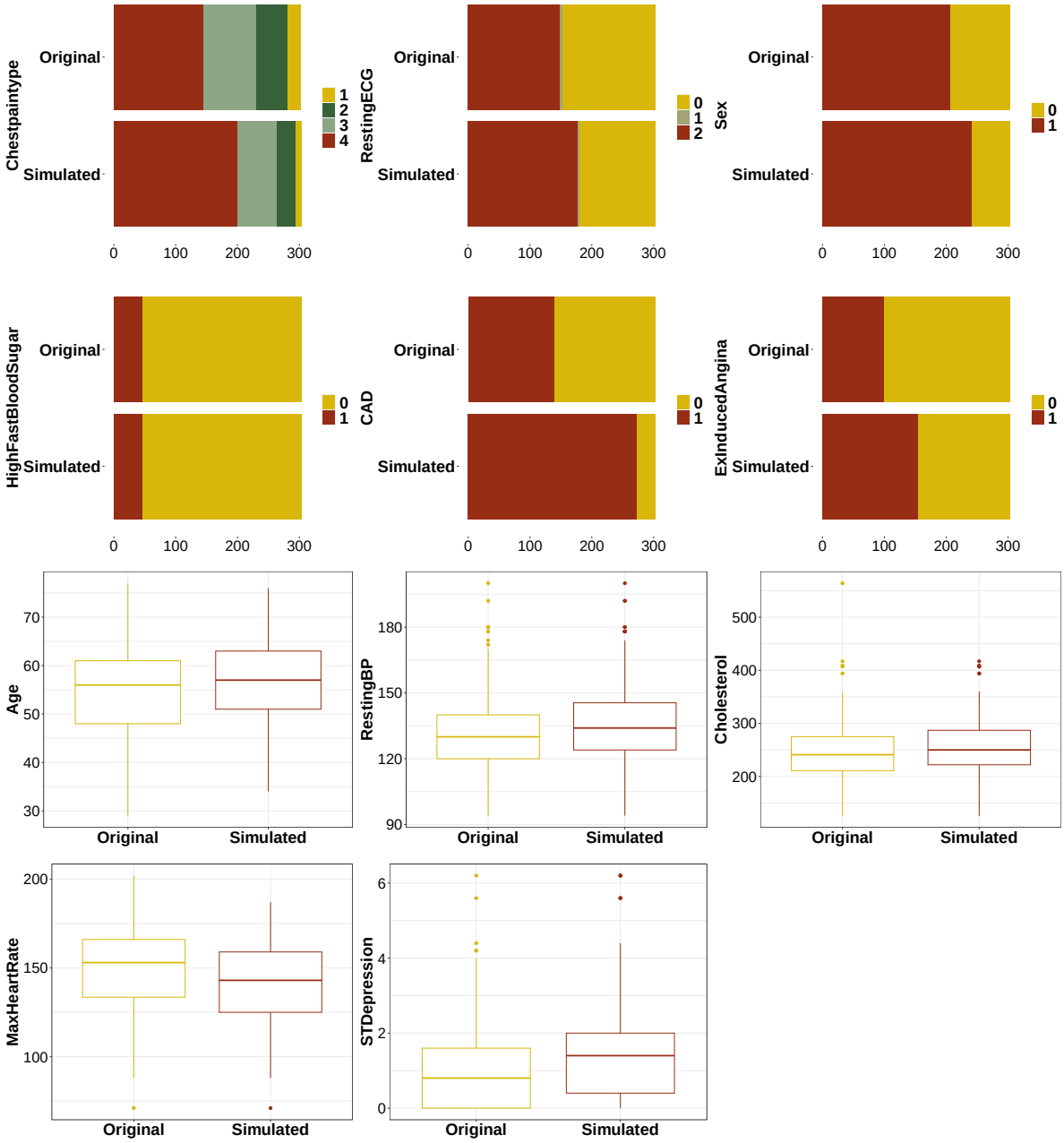


Figure 14: Distribution plot to depict the distribution for each variable of the original data set and a single simulated data set when the proportion of subjects with coronary artery disease was set to 90%.



6 Expansion 4: Multicenter analysis

Centers in multicenter studies may be heterogeneous, thus different in their structure. This can be addressed in *modgo* as follows. First, the user may conduct multiple *modgo* runs, one for each center. Second, the different simulated data sets can be merged with the *modgo* function *multicenter_comb*.

This expansion to multicenter studies is demonstrated using the four centers Cleveland, Swiss, Hungarian, and Veterans, which are provided by Cleveland Clinic data set (Dua and Graff 2017). We compared this multicenter run with a *modgo* simulation that ignored the multicenter structure of the Cleveland Clinic data set. We refrained from including cholesterol levels in this illustration because the Swiss data set did not have this variable.

Figures 15 through 20 show the correlation plots for the Cleveland Clinic data set. Figure 15 displays results for the data from the Cleveland Clinic, Figure 16 for the Swiss data, Figure 17 for the Hungarian data and Figure 18 for the Veterans data. Furthermore, Figure 19 and Figure 20 provide the correlation plots, when the multicenter nature of the data was not taken and was taken into account, respectively. Differences between the original and the simulated data set were small. However, in the main article we provide results from logistic regression analysis, for which we indicate an advantage of the multicenter simulation approach over the method ignoring the multicenter nature of the data.

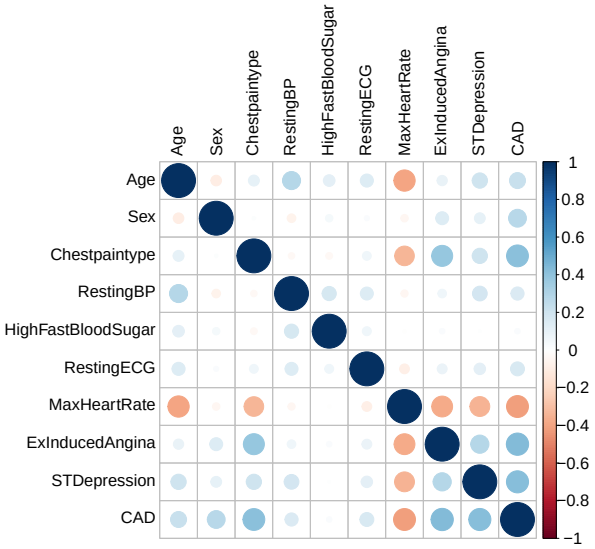
Figures 21 through 26 show the distribution plots for the multicenter expansion with the Cleveland Clinic data set. Specifically, the Cleveland Clinic, Swiss, Hungarian and Veterans data are given in Figure 21, Figure 22, Figure 23, and Figure 24, respectively. Figure 25 provides the distribution plots when the multicenter nature of the data was ignored. Finally, Figure 26 shows the distribution plots, when the multicenter setting was taken into account in the simulations.

```
#Create named vector

test_Cleve <-  modgo(data = Cleve[,-c(5,12)],bin_variables = binary_variables,
                 categ_variables = categorical_variables,nrep = nrep)

test_Swiss <-  modgo(data = Swiss[,-c(5,12)],bin_variables = binary_variables,
                 categ_variables = categorical_variables,nrep = nrep)

test_Hung <-  modgo(data = Hung[,-c(5,12)],bin_variables = binary_variables,
                 categ_variables = categorical_variables,nrep = nrep)

test_VA <-  modgo(data = VA[,-c(5,12)],bin_variables = binary_variables,
                 categ_variables = categorical_variables,nrep = nrep)
```

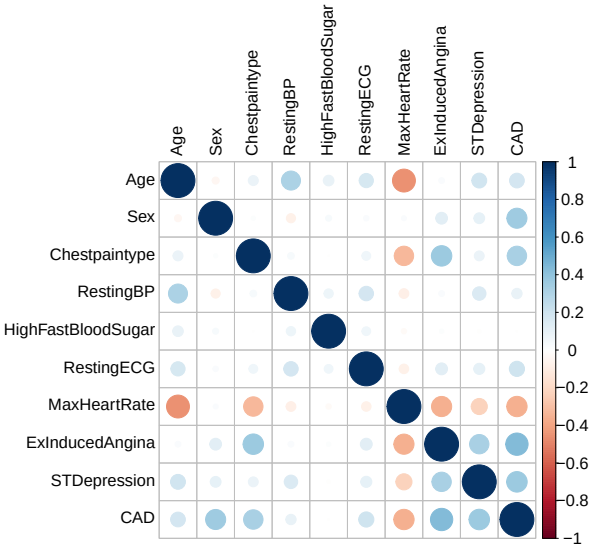
```
combined_dataset <- rbind(Cleve[,-c(5,12)], Swiss[,-c(5,12)],
                          Hung[,-c(5,12)], VA[,-c(5,12)])

test_whole_dataset <- modgo(data = combined_dataset,
                            bin_variables = binary_variables,
                            categ_variables = categorical_variables,nrep = nrep)

test_multicent <- multicenter_comb(modgo_1 = test_Cleve, modgo_2 = test_Swiss,
                                   modgo_3 = test_Hung, modgo_4 = test_VA)
```

Figure 15: Correlation plots for a *modgo* run for the data from the Cleveland Clinic. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

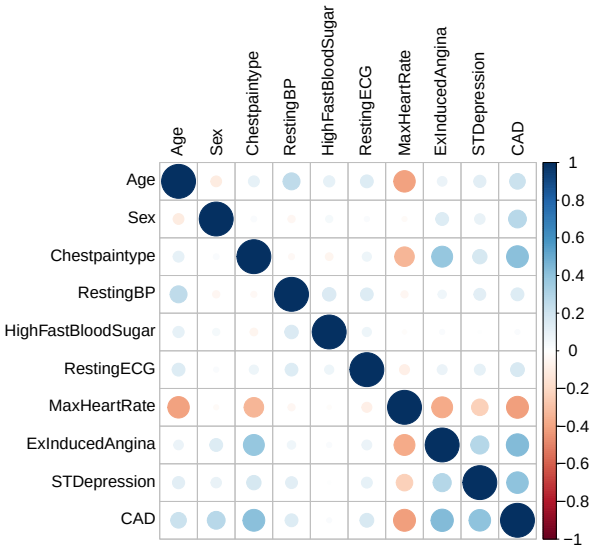
Original



Simulated



Mean correlation of simulations



Original minus simulated

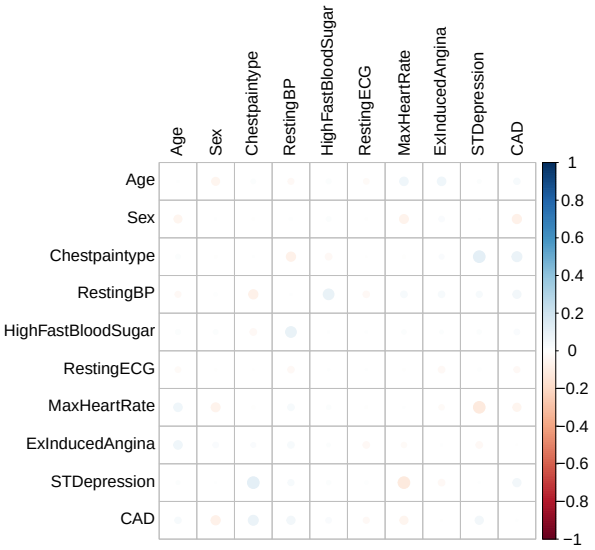
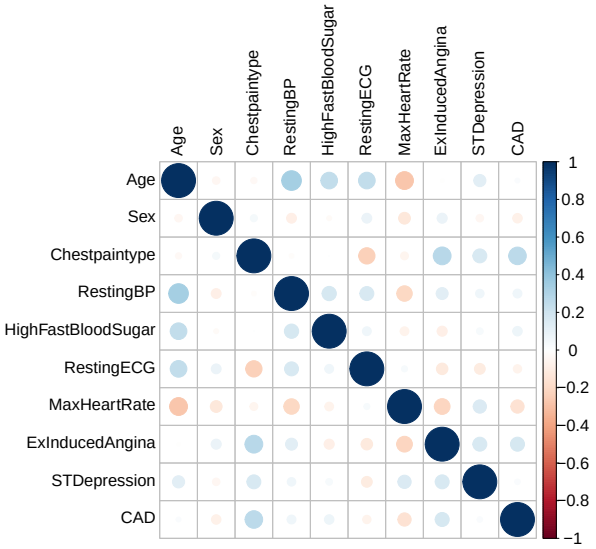
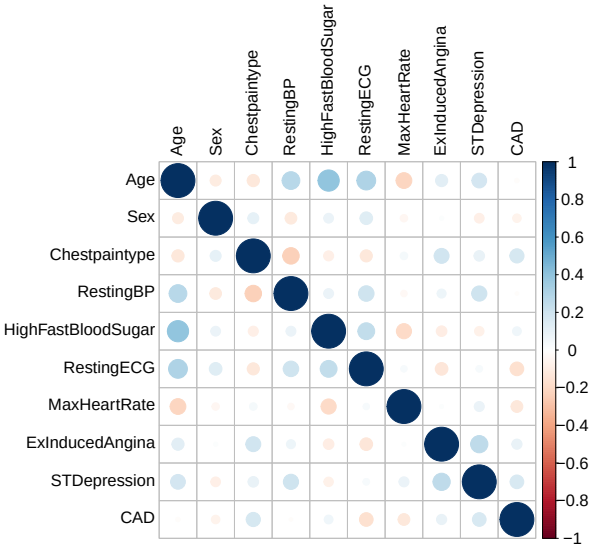


Figure 16: Correlation plots for a *modgo* run for the Swiss data set. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

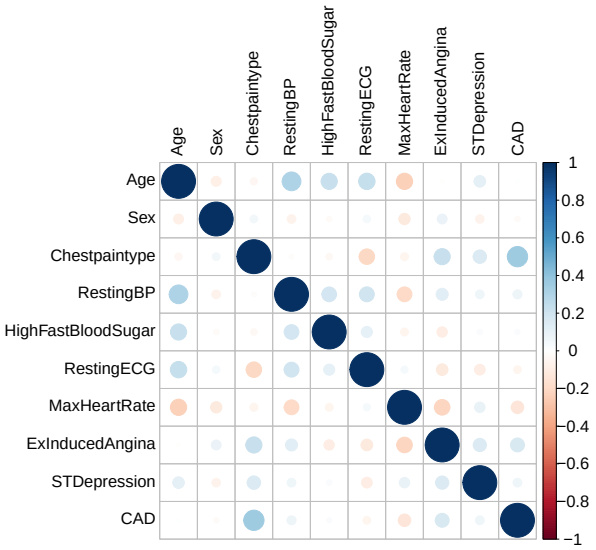
Original



Simulated



Mean correlation of simulations



Original minus simulated

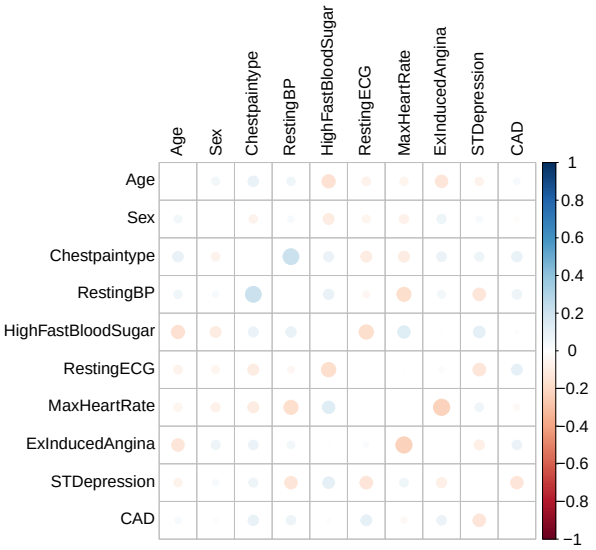
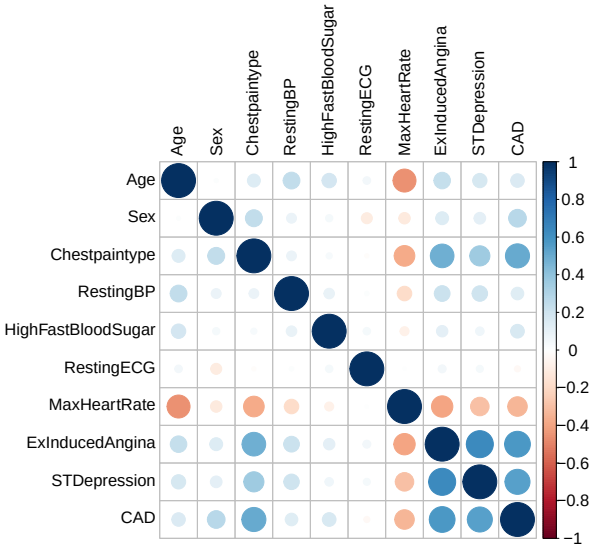
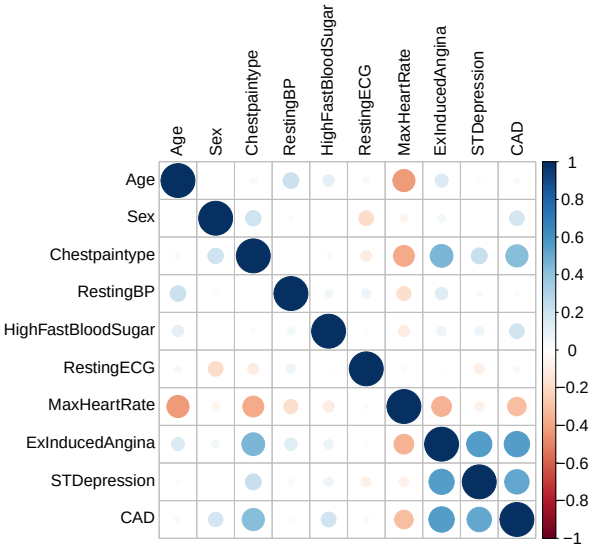


Figure 17: Correlation plots for a *modgo* run for the Hungarian data set. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

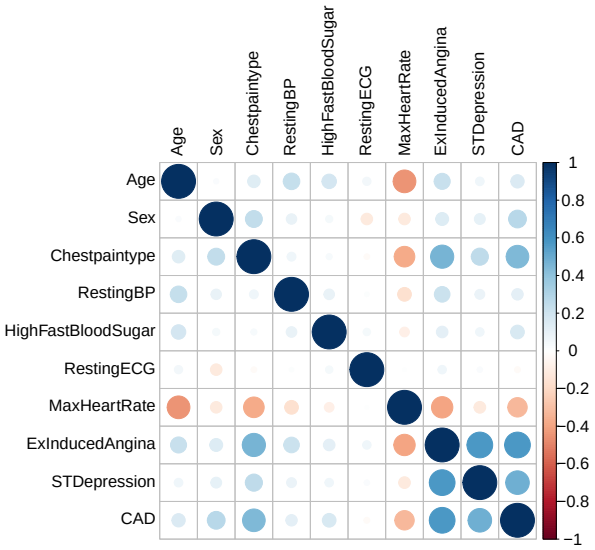
Original



Simulated



Mean correlation of simulations



Original minus simulated

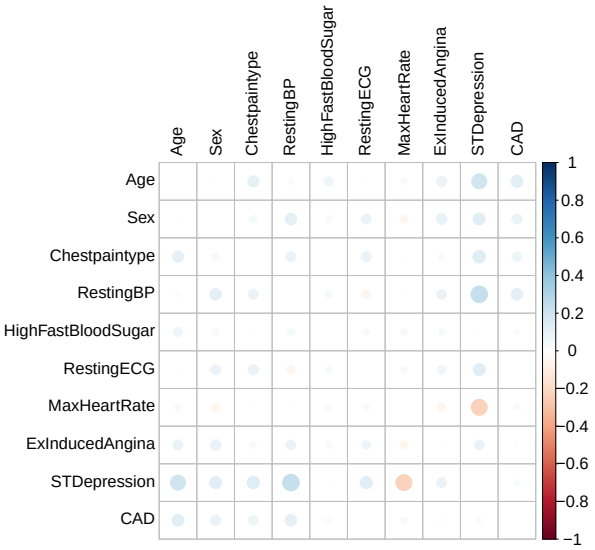
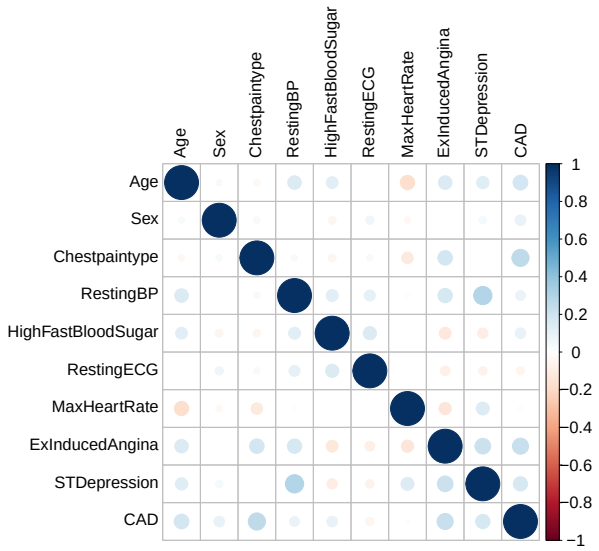
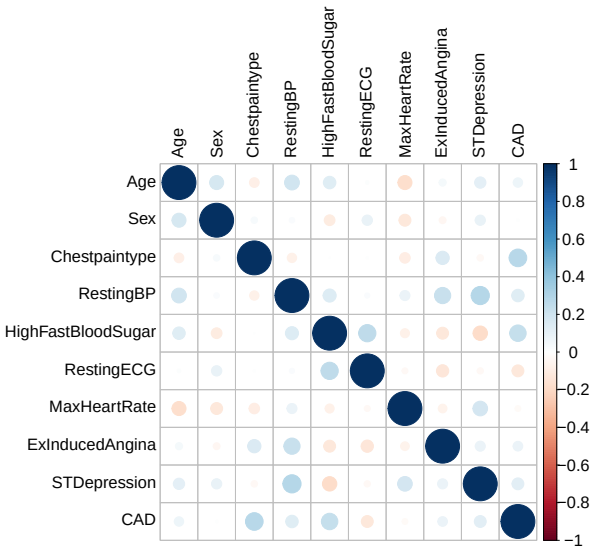


Figure 18: Correlation plots for a *modgo* run for the Veterans data set. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

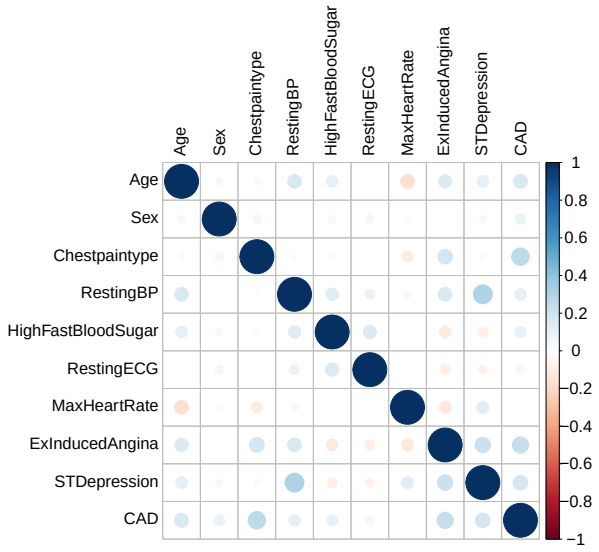
Original



Simulated



Mean correlation of simulations



Original minus simulated

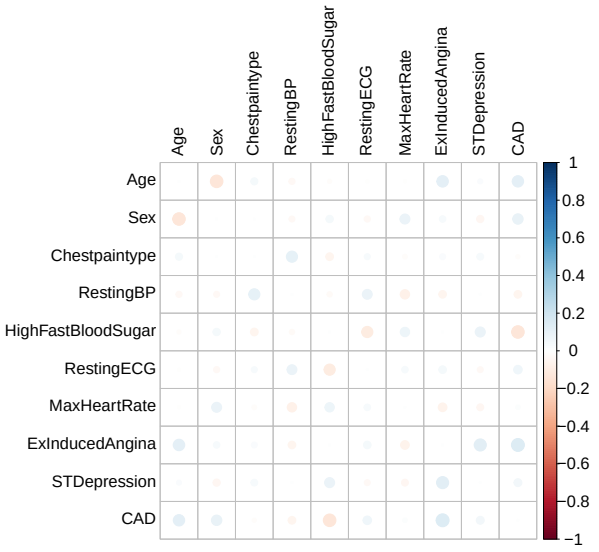
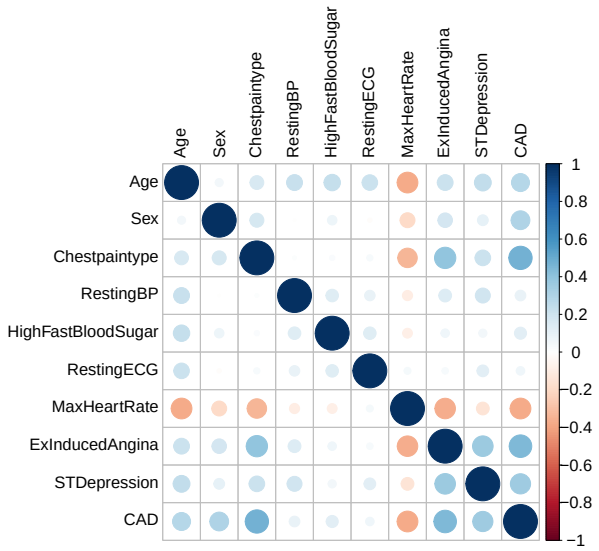
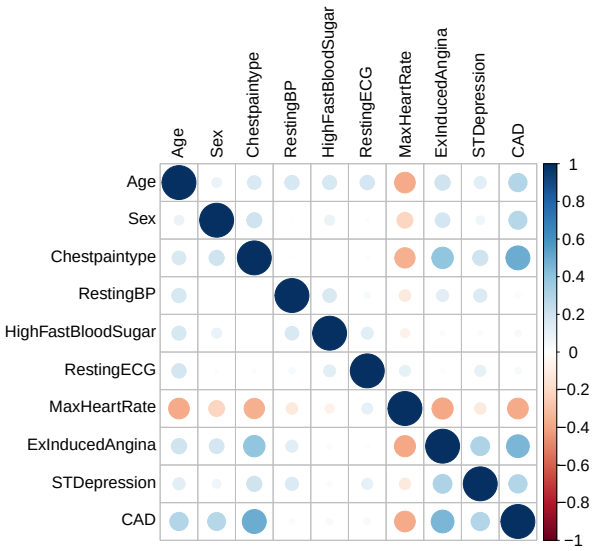


Figure 19: Correlation plots for a *modgo* run for the Cleveland Clinic data set, when the multicenter nature of the data was ignored. Data were thus assumed to be homogeneous. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

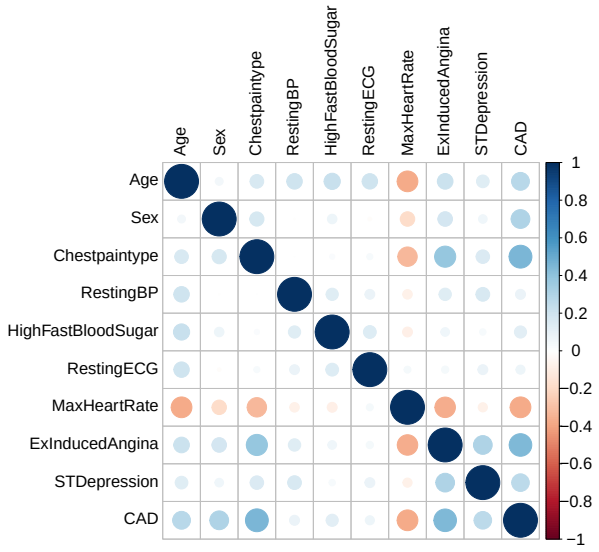
Original



Simulated



Mean correlation of simulations



Original minus simulated

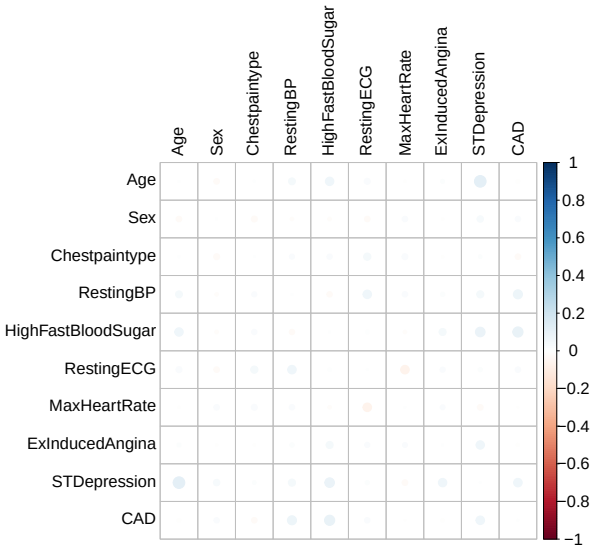
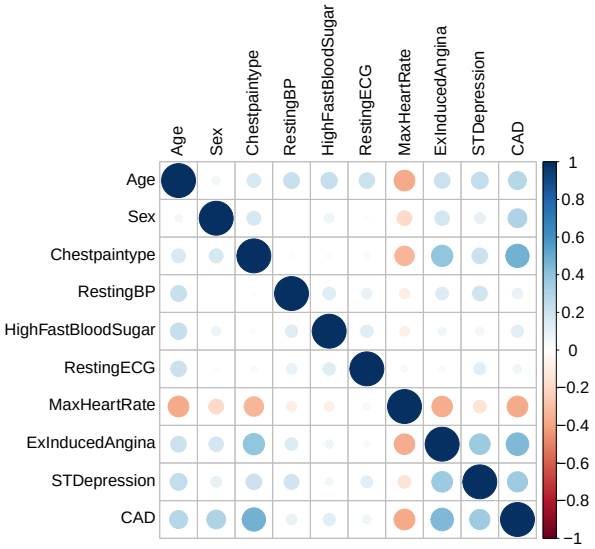
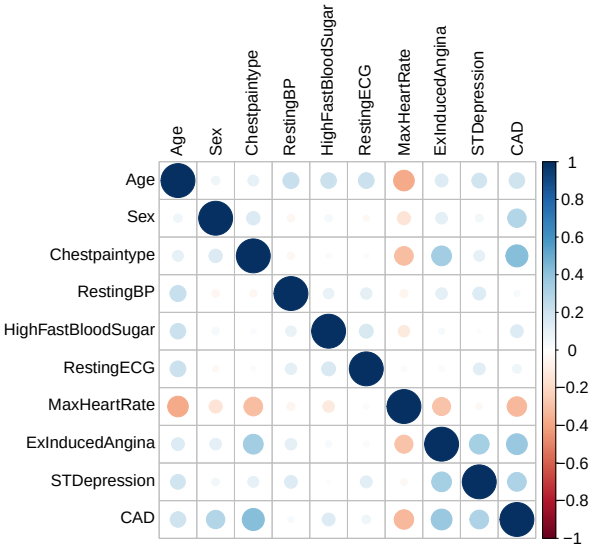


Figure 20: Correlation plots for a *modgo* run for the Cleveland Clinic data set, when the multicenter nature of the data was taken into account. Displayed are the correlations of the original data set, the correlations of a single simulated data set, the mean correlation matrix over all simulated data sets, and the difference between the correlations of the original data set and a single simulated data set.

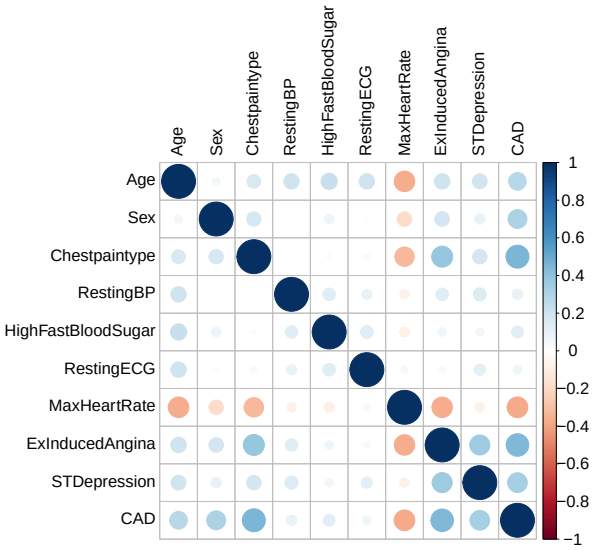
Original



Simulated



Mean correlation of simulations



Original minus simulated

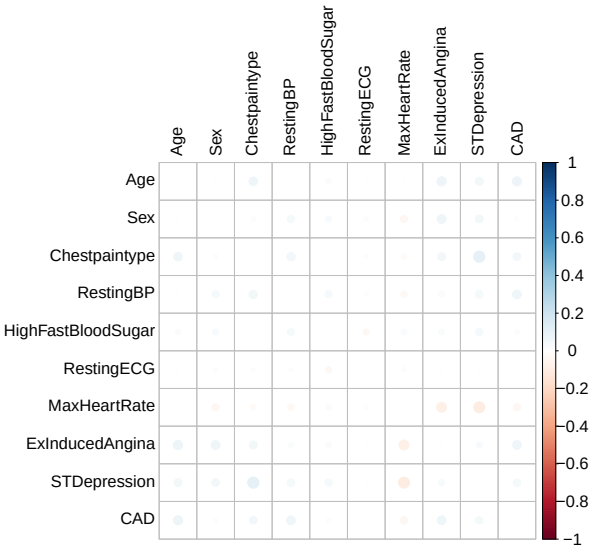


Figure 21: Distribution plots for the data set from the Cleveland Clinic.

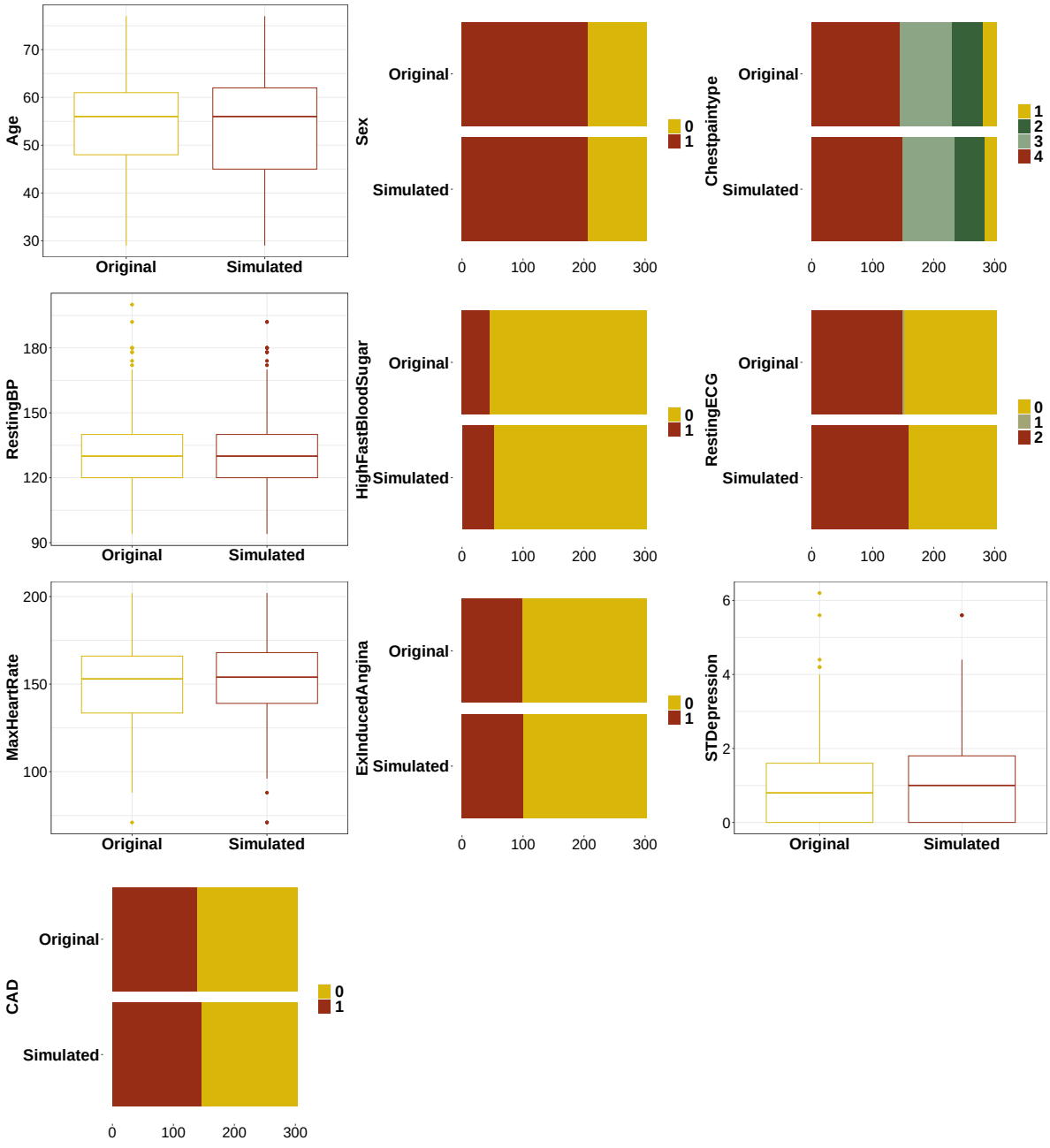


Figure 22: Distribution plot for the Swiss data set.

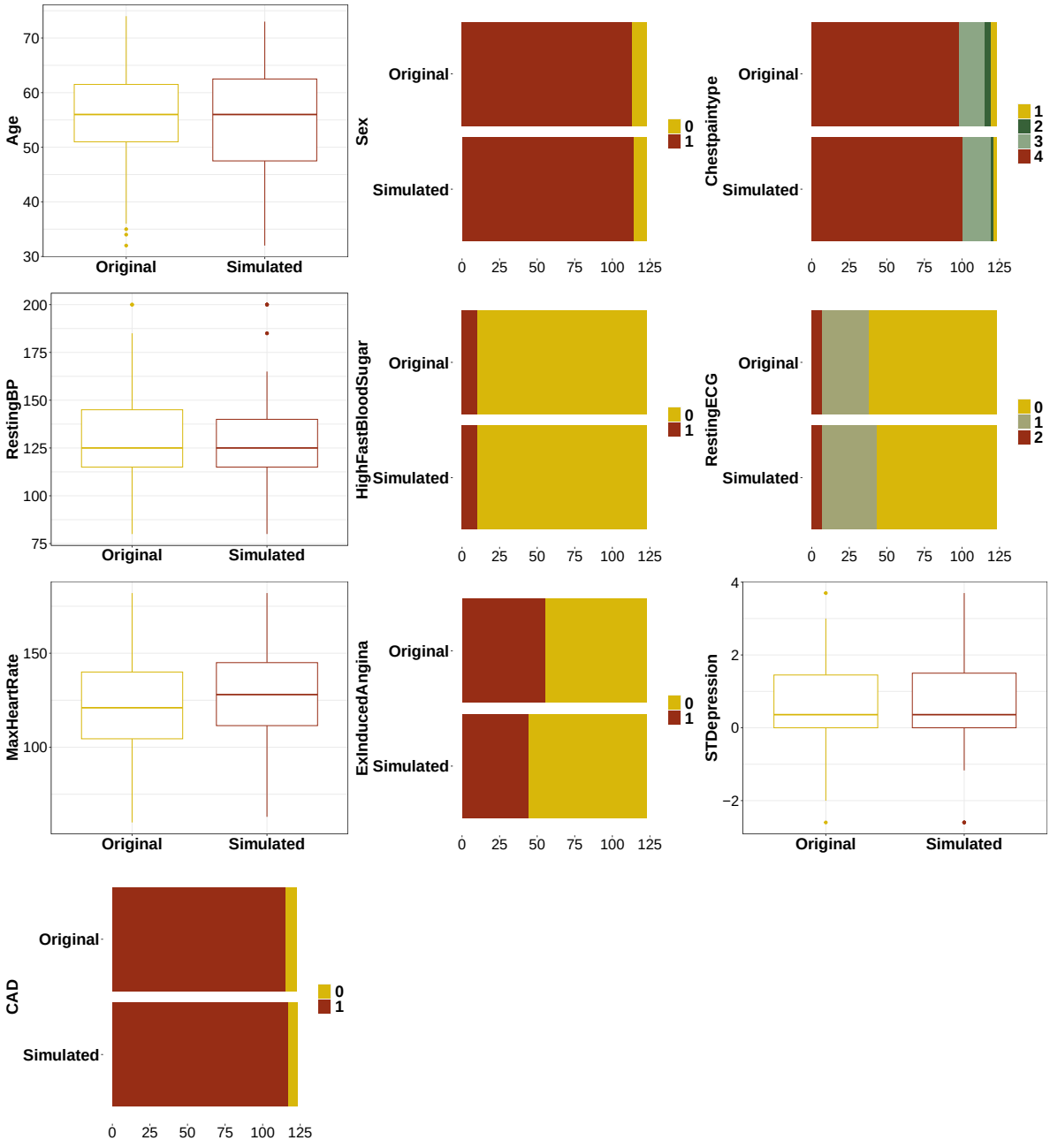


Figure 23:Distribution plot for the Hungarian data set.

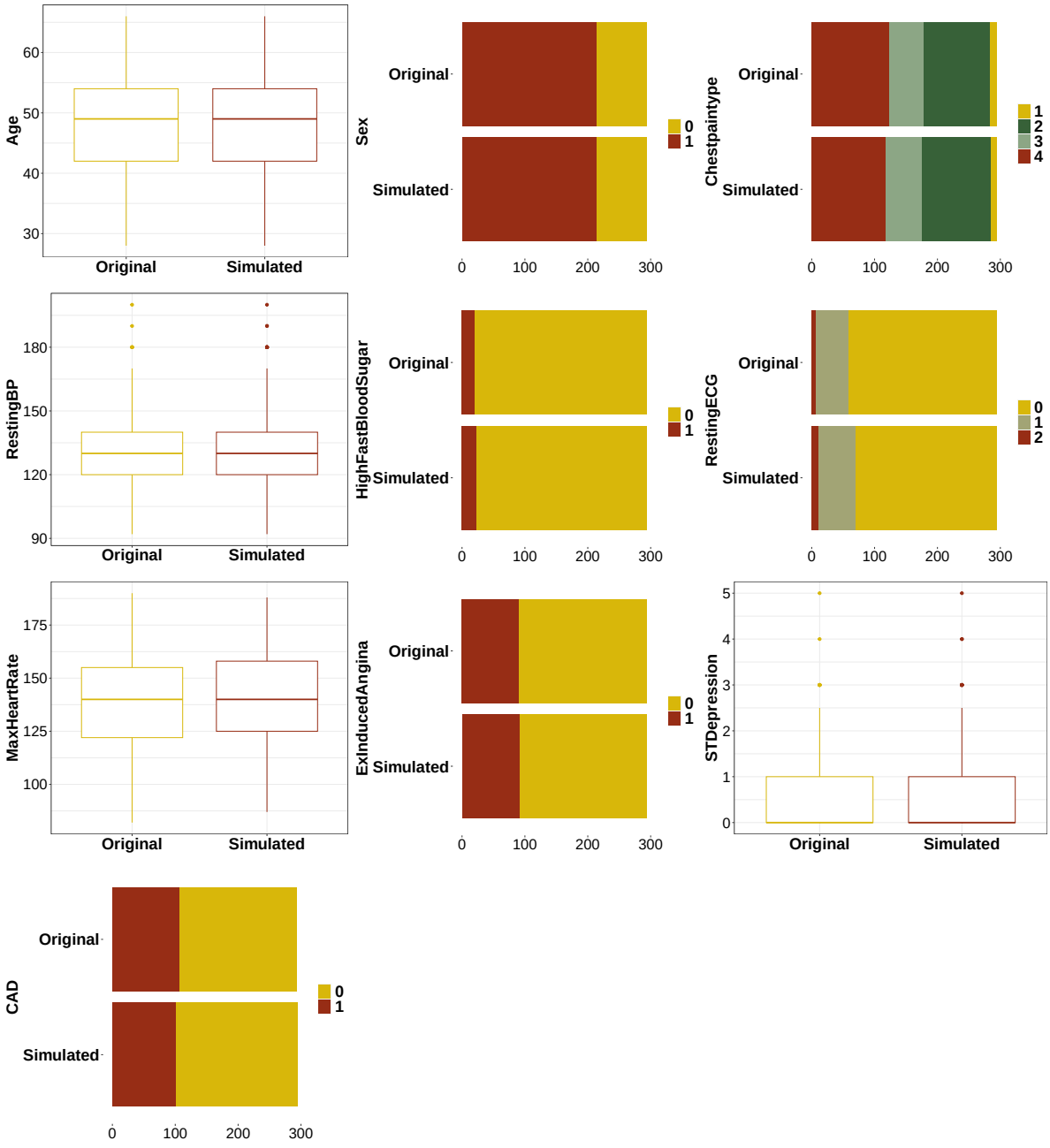


Figure 24: Distribution plot for the Veterans data set.

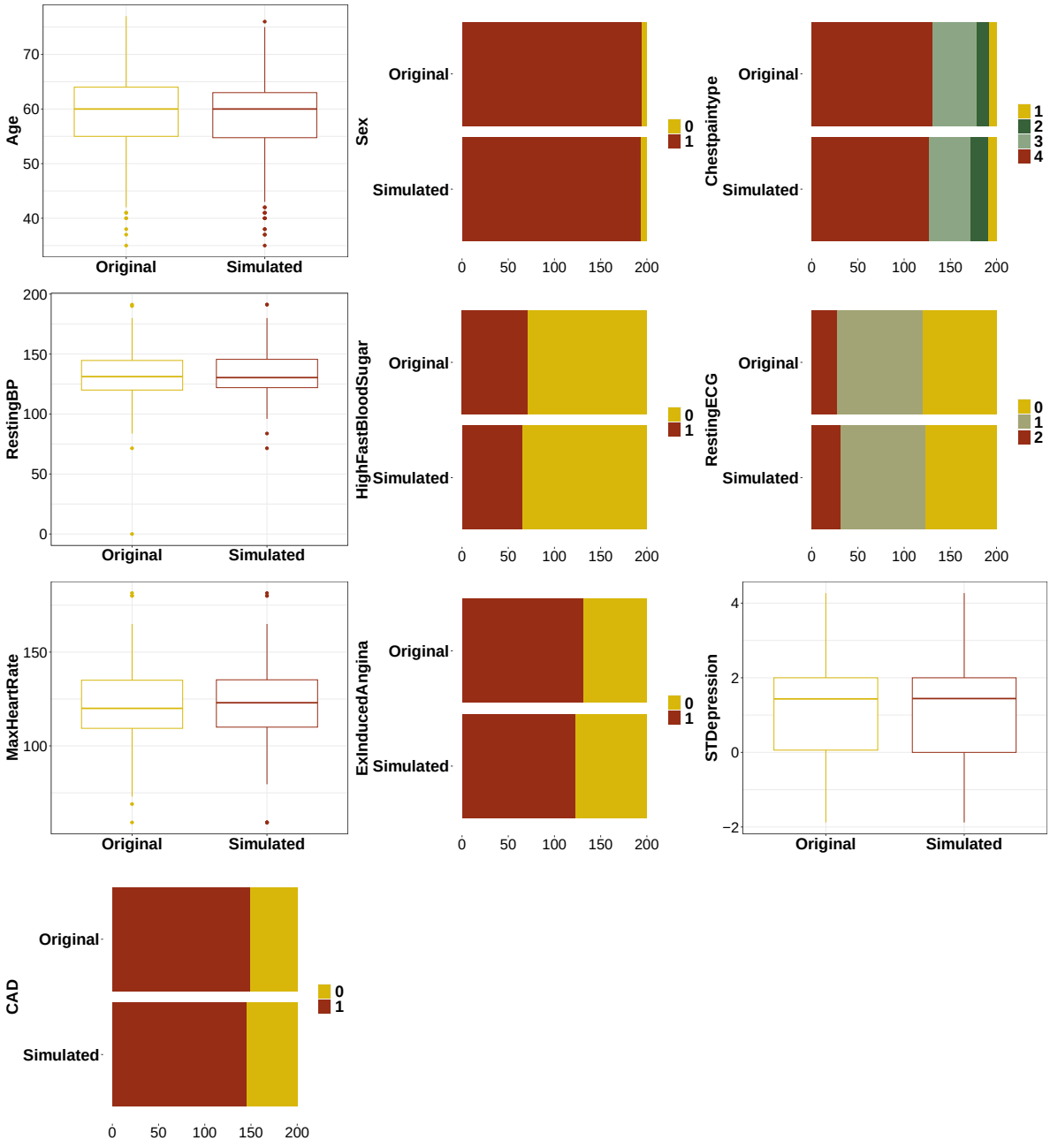


Figure 25: Distribution plot for the combination of all four data sets from the Cleveland Clinic Data set, when the multicenter nature of the study was ignored in the simulations.

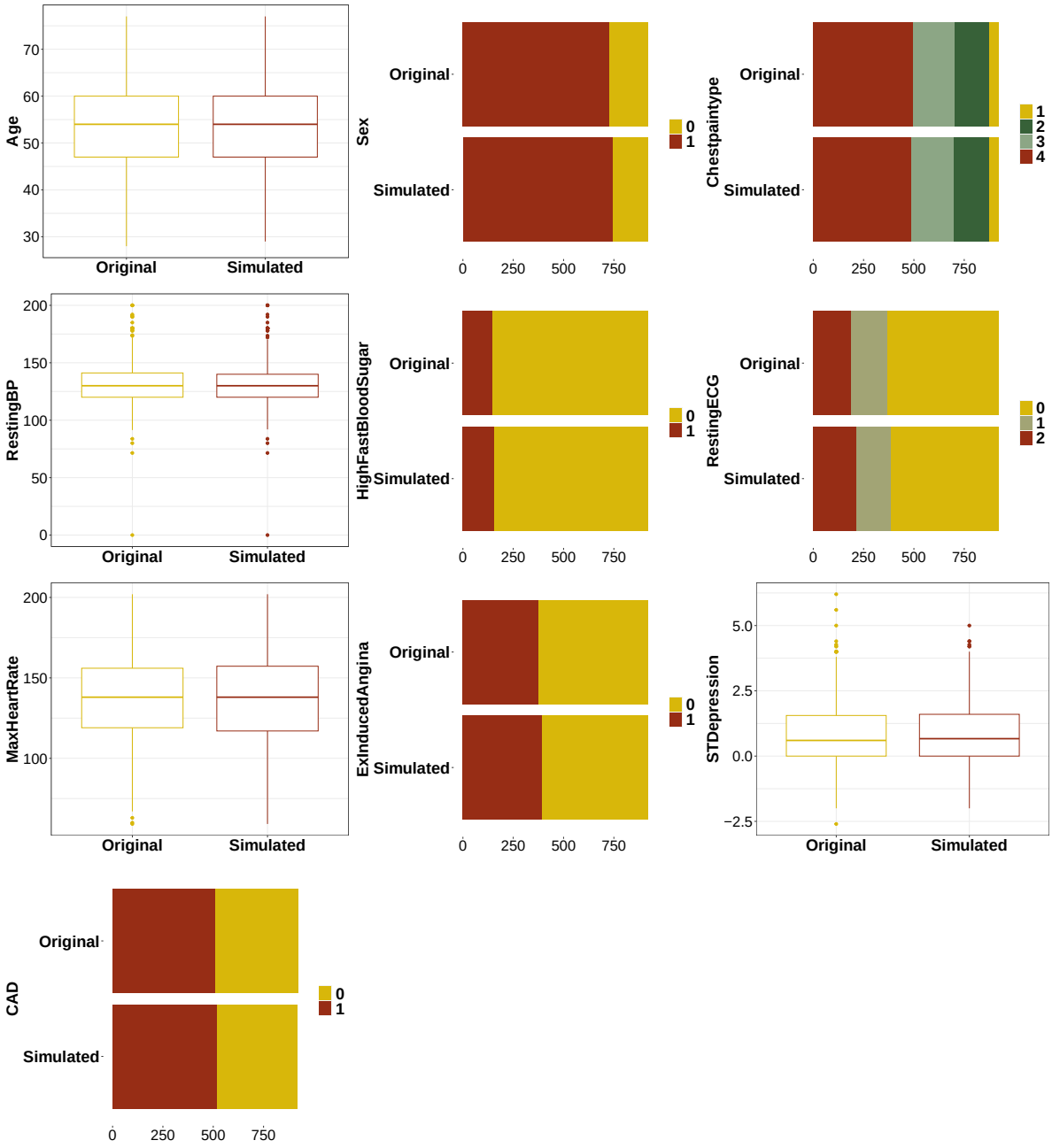
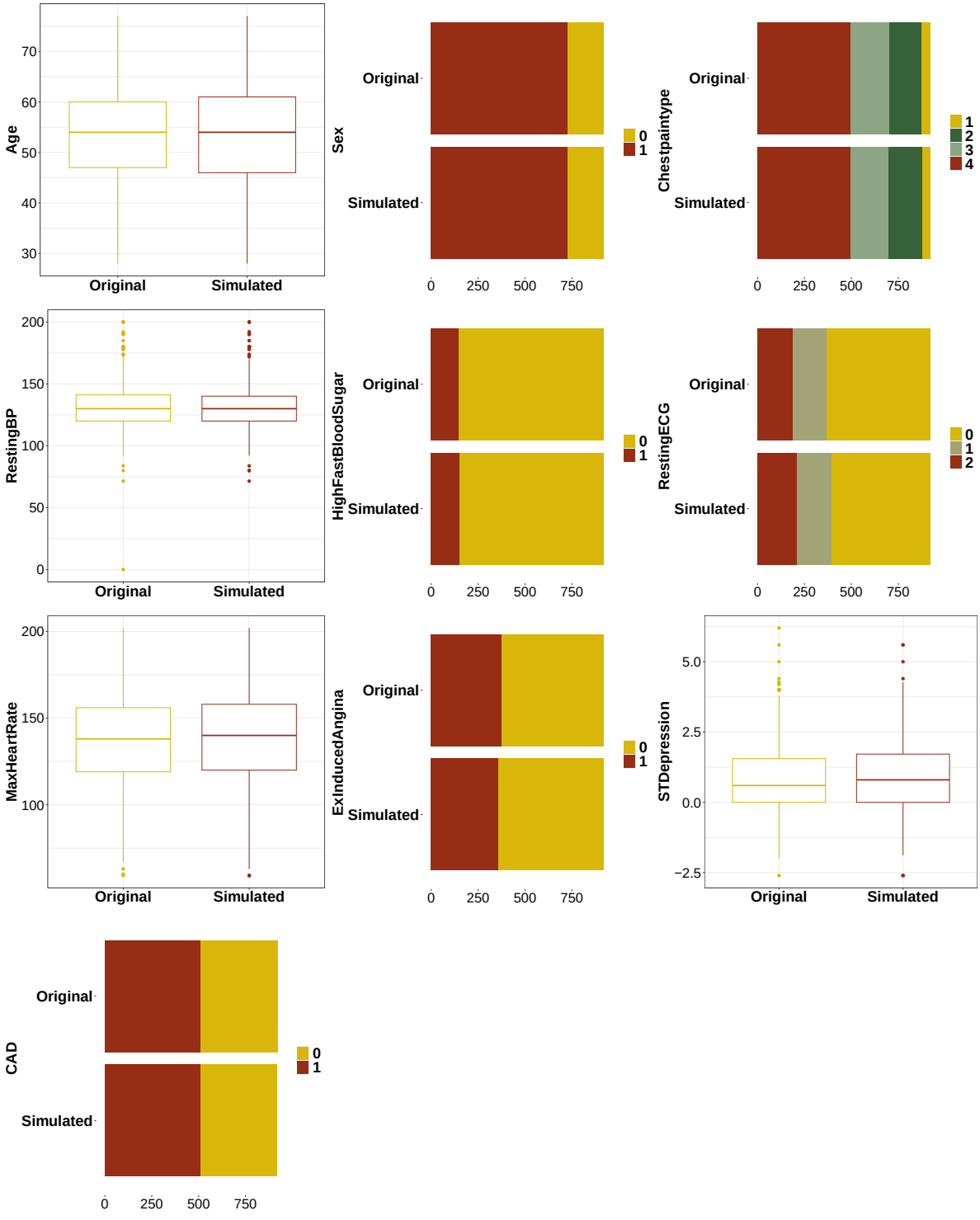


Figure 26: Distribution plot for the combination of all four data sets from the Cleveland Clinic Data set, when the multicenter nature of the study was taken into account in the simulations.


Age
70
60
50
40
30
Original
Simulated
Sex
Original
Simulated
0
1
0
250
500
750
Chestpaintype
Original
Simulated
1
2
3
4
0
250
500
750
RestingBP
200
150
100
50
0
Original
Simulated
HighFastBloodSugar
Original
Simulated
0
1
0
250
500
750
RestingECG
Original
Simulated
0
1
2
0
250
500
750
MaxHeartRate
200
150
100
Original
Simulated
ExInducedAngina
Original
Simulated
0
1
0
250
500
750
STDepression
5.0
2.5
0.0
−2.5
Original
Simulated
CAD
Original
Simulated
0
1
0
250
500
750

7 Logistic regression results

In this section, logistic regression odds ratios are displayed for the original data and for *modgo* runs in the default mode and with different expansions. Coronary artery disease (CAD) was used as dependent variable, and, for illustration, exercise-induced angina, age, max heart rate, and ST depression were used as independent variables.

Table 1: Odds ratio estimates for the Cleveland Clinic Data. Displayed are estimates from: Table 1.a: The original data (original), the default *modgo* run (default), when subjects had to be at least 66 years old (age thresh, When thresholds were introduced for three continuous variables as described above (3 variables thresh). Table 1.b: When data were perturbed with proportion 90% for resting blood pressure and 70% for cholesterol while keeping the variance unchanged, and when the proportion of subjects with coronary artery disease had to be at least 90%. Odds ratios are displayed for the original data, median odds ratios and empirical 2.5% and 97.5% quantiles for the odds ratio estimates are displayed in parenthesis for each variable included in the logistic regression. Results are shown for 500 simulated data sets for each *modgo* run.

Table 1.a

	Original	Default	Age thresh	3 variables thresh
Exercise induced angina (yes)	4.43	4.45 (2.38 - 8.17)	4.65 (2.28 - 11.8)	4.39 (2.33 - 9.28)
Age (years)	1.02	1.02 (0.98 - 1.05)	1.03 (0.92 - 1.16)	1.02 (0.91 - 1.14)
Max heart rate (bpm)	0.98	0.97 (0.95 - 0.99)	0.97 (0.96 - 0.99)	0.97 (0.96 - 0.98)
ST depression (mm)	1.92	2.07 (1.52 - 2.86)	2.07 (1.51 - 2.95)	2.04 (1.55 - 2.82)

Table 1.b

	Original	Perturb(Unchanged Variance)	CAD prop
Exercise induced angina (yes)	4.43	4.24 (2.32 - 8.13)	4.51 (1.67 - 29.31)
Age (years)	1.02	1.02 (0.99 - 1.06)	1.02 (0.97 - 1.08)
Max heart rate (bpm)	0.98	0.97 (0.95 - 0.99)	0.97 (0.94 - 0.99)
ST depression (mm)	1.92	2.09 (1.57 - 2.79)	2.22 (1.4 - 4.79)

Table 2: Odds ratio estimates for the Cleveland Clinic Data. Displayed are estimates from the original data (original), the *modgo* run, when the multicenter nature of the data was ignored, i.e., all centers were treated as one (As one data set), and when the multicenter nature of the data was taken into account (Multicenter) . Odds ratios are displayed for the original data, median odds ratios and empirical 2.5% and 97.5% quantiles for the odds ratio estimates are displayed in parenthesis for each variable included in the logistic regression. Results are shown for 500 simulated data sets for each *modgo* run.

	All original data	As One dataset	Multicenter
Exercise induced angina (yes)	3.8	4.37 (3.13 - 6.1)	3.83 (2.66 - 5.73)
Age (years)	1.03	1.03 (1.02 - 1.05)	1.03 (1.01 - 1.05)
Max heart rate (bpm)	0.98	0.98 (0.97 - 0.98)	0.98 (0.97 - 0.98)
ST depression (mm)	1.7	1.52 (1.29 - 1.78)	1.78 (1.5 - 2.12)

References

Dua, Dheeru, and Casey Graff. 2017. "UCI Machine Learning Repository." University of California, Irvine, School of Information; Computer Sciences. http://archive.ics.uci.edu/ml.