Missing Data Imputation

Ajinkya Kothavale

12/1/2024

Introduction:

The issue of missing data is a common challenge in data analysis, and handling such missingness is crucial for the validity and reliability of statistical models. In this project, we focus on imputing missing values in a dataset collected from a study of diabetes among African American subjects. The dataset contains various variables, including demographic and health-related features, with some values missing. This report investigates several imputation methods, including complete case analysis, mean imputation, and multiple imputation via chained equations (MICE), to address missing data. The goal is to compare the performance of these imputation techniques by fitting linear models to the imputed datasets and evaluating the resulting standard errors and coefficient estimates. The analysis provides insight into the accuracy and reliability of imputation methods in handling missing values in a real-world dataset.

Objectives:

The primary objectives of this analysis are:

1. To explore the extent and nature of missing data in the diabetes dataset using exploratory data analysis (EDA) techniques such as visualizations and summary statistics.
2. To apply different imputation methods (complete case analysis, mean imputation, and multiple imputation by chained equations) to handle missing data.
3. To compare the results of these imputation methods by fitting linear regression models and examining the estimates and standard errors of the coefficients.
4. To assess the performance of multiple imputation using MICE through diagnostic plots and comparisons of the imputation methods based on pooled results.
5. To quantify the variability between imputations and evaluate the fraction of information lost due to missing data, providing insights into the effectiveness of imputation strategies.

Methods Detail:

This project follows a structured approach to handle missing data, beginning with exploratory data analysis (EDA) and progressing through various imputation techniques. The methods used are described in detail as follows:

Data Preprocessing:

The dataset is first read into R and cleaned by removing the stabilized glucose variable, as it directly indicates the onset of diabetes and is excluded from imputation models. The extent and pattern of missingness are analyzed using functions like aggr() and matrixplot() from the VIM package to visualize missing data patterns.

Correlation Analysis:

A correlation plot is generated using the corrplot() function to examine relationships between variables in the dataset, helping inform the imputation process.

Imputation Methods:

1. Complete Case Analysis: This method involves analyzing only the cases with no missing data. The analysis is performed by removing rows with missing values (na.omit()), and linear regression models are fitted to the complete data.

2. Mean Imputation: Missing values are replaced by the mean of the observed values for each variable. The imputation is performed using the mice() function with the method set to “mean.” Linear regression models are then fitted to the imputed dataset.

3. Multiple Imputation by Chained Equations (MICE): The mice() function is also used for multiple imputation, where 20 imputed datasets are generated. Linear regression models are fitted to each imputed dataset, and the results are pooled using the pool() function. This method accounts for uncertainty in the missing data and provides more reliable estimates than single imputation methods.

Model Comparison:

Linear models are fitted on the imputed datasets, and the resulting estimates and standard errors are compared across the three imputation methods (complete case analysis, mean imputation, and multiple imputation).

Diagnostic plots (density plots, strip plots, etc.) are used to assess the distributions of the imputed values and the overall convergence of the MICE algorithm.

Model comparison is performed using the pool.compare() function to assess the differences between models fitted on the various imputed datasets.

Assessment of Variability:

The variability within and between imputations is calculated by examining the coefficients and variances of the imputed datasets. The total variability is obtained by combining the within-imputation and between-imputation variances. The fraction of information lost due to missing data is also computed. Visualization and Results:

Bar plots, density plots, and box plots are used to visually compare the results of the three imputation methods in terms of coefficient estimates and standard errors.

The analysis concludes with a comparison of the estimates and standard errors from the different imputation methods and a final interpretation of the results.

Step 1: Setup Libraries

```
# Install necessary packages (uncomment if not installed)
# install.packages(c("VIM", "corrplot", "mice", "xtable", "Matrix"))

# Load libraries
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 4.3.3
```

```
## Warning: package 'colorspace' was built under R version 4.3.3
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.3
```

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 4.3.3
```

```
## Warning in check_dep_version(): ABI version mismatch:
## lme4 was built with Matrix ABI version 1
## Current Matrix ABI version is 0
## Please re-install lme4 from source or restore original 'Matrix' package
```

```
library(xtable)
```

```
## Warning: package 'xtable' was built under R version 4.3.3
```

```
library(Matrix)
```

Step 2: Loading the Data

```
# Set seed for reproducibility
set.seed(1234)

# Import the dataset
DM = read.csv("C:/Users/Ajinkyaa/OneDrive/Stata to R/New folder/Missing data analysis/Missing
-Data-Imputation/diabetes_C.csv")
```

Step 3: Preprocessing and Initial Exploration

```
# Remove 'stab.glu' column as it indicates diabetes onset
DM = DM[,-c(2)]

# Display dataset structure
names(DM)
```

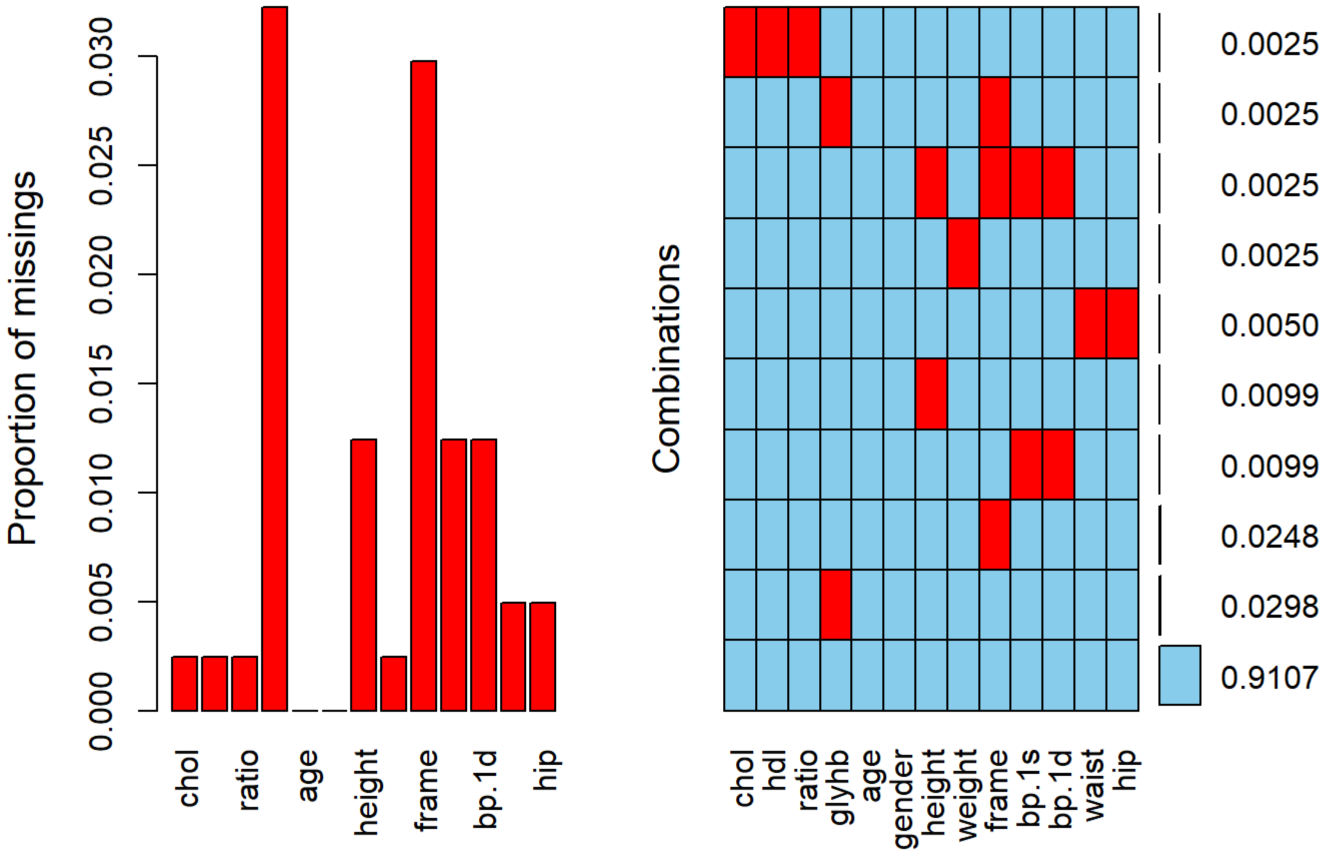
```
##  [1] "chol"   "hdl"    "ratio"  "glyhb"  "age"    "gender" "height" "weight"
##  [9] "frame"  "bp.1s"  "bp.1d"  "waist"  "hip"
```

```
head(DM)
```

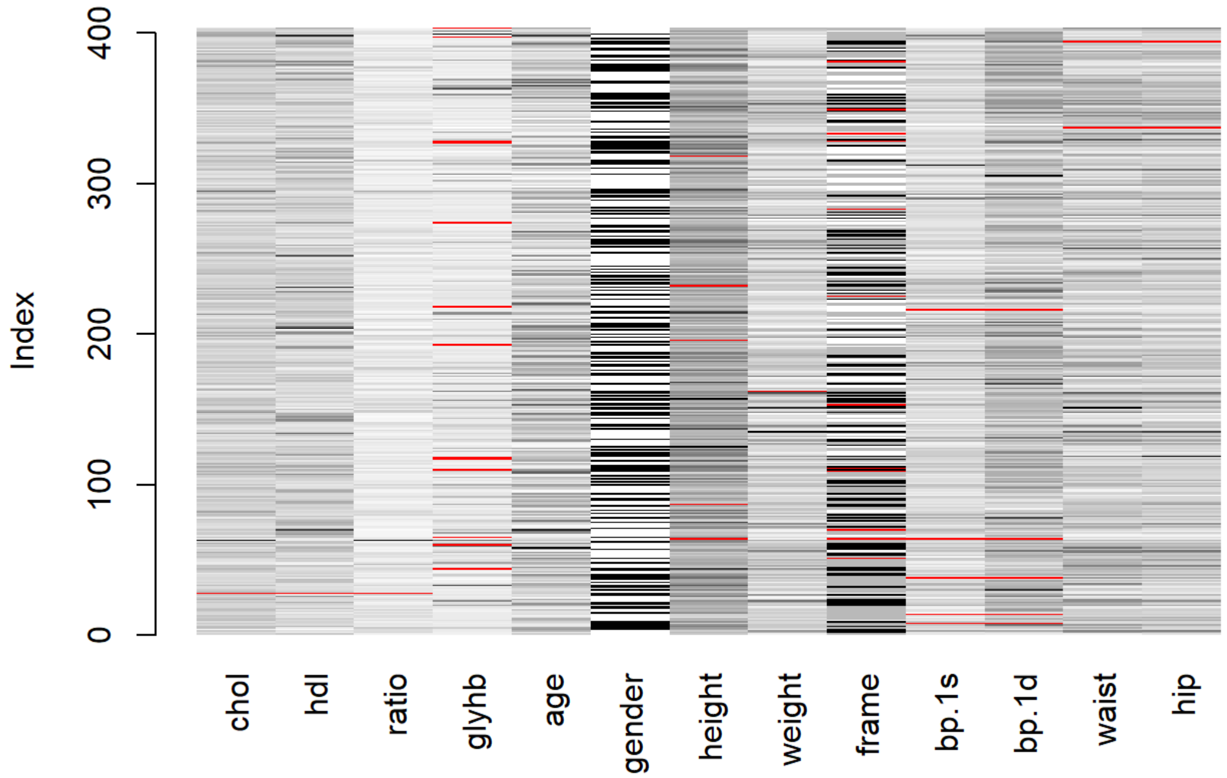
```
##   chol hdl ratio glyhb age gender height weight frame bp.1s bp.1d waist hip
## 1  203  56   3.6  4.31  46 female     62    121     2   118    59    29  38
## 2  165  24   6.9  4.44  29 female     64    218     3   112    68    46  48
## 3  228  37   6.2  4.64  58 female     61    256     3   190    92    49  57
## 4   78  12   6.5  4.63  67   male     67    119     3   110    50    33  38
## 5  249  28   8.9  7.72  64   male     68    183     2   138    80    44  41
## 6  248  69   3.6  4.81  34   male     71    190     3   132    86    36  42
```

Step 4: Missing Data Visualization

```
# Visualize missing data
aggr(DM, numbers = TRUE, main = "Missing Data Overview")
```

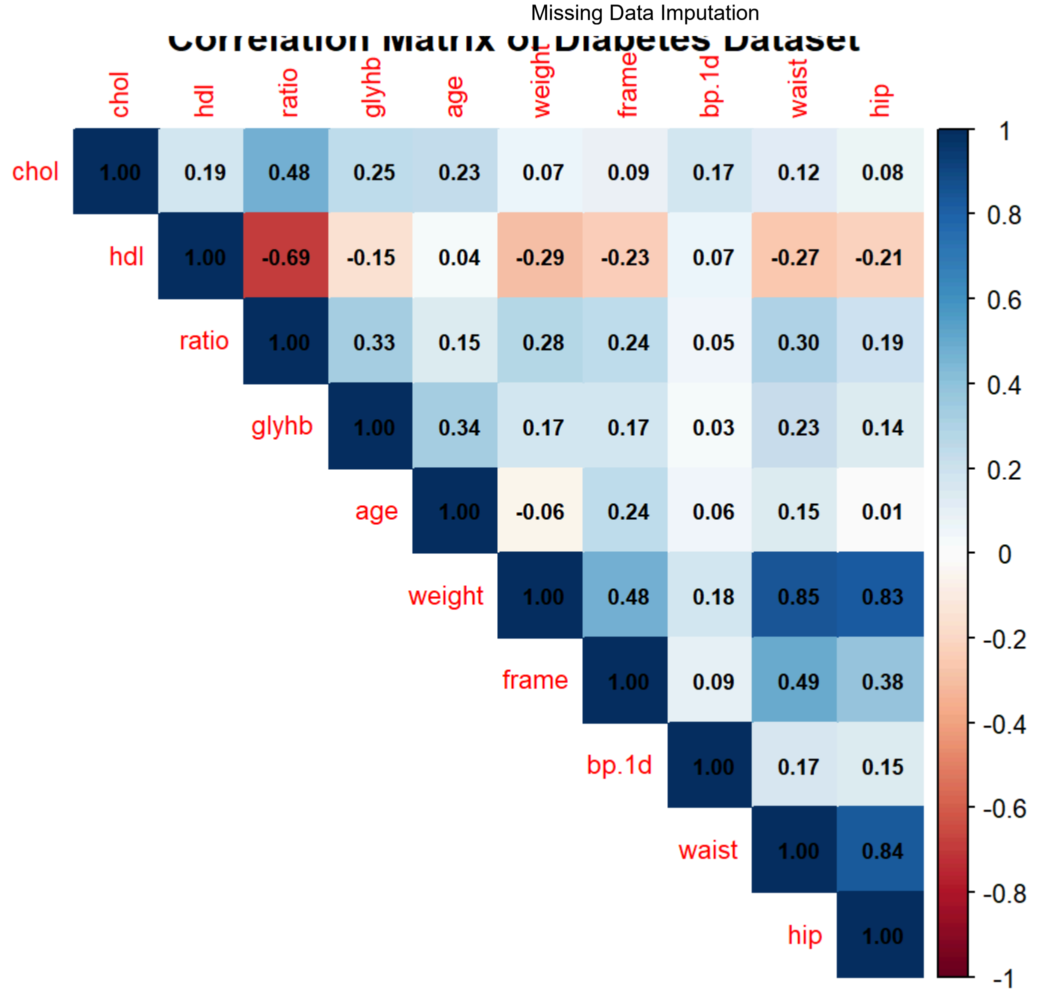


```
matrixplot(DM) # Identify patterns in missingness
```



Step 5: Correlation Analysis

```
# Extract numeric data (excluding columns 6 and 9)
numeric_data <- DM[, sapply(DM, is.numeric)]

# Compute the correlation matrix
cor_matrix <- cor(numeric_data[, -c(6, 9)], use = "pairwise.complete.obs")

# Plot Correlation Heatmap
corrplot(cor_matrix,
         method = "color",
         type = "upper",
         addCoef.col = "black",
         tl.cex = 0.8,
         number.cex = 0.7,
         main = "Correlation Matrix of Diabetes Dataset")
```



Step 6: Complete Case Analysis

```
# Retain only rows with complete data
DM1 = na.omit(DM)
y = DM1[,4]  # Response variable
X = DM1[,-4] # Predictor variables

# Fit a linear model
Model1 = lm(y ~ ., data = X)
Model2 = step(Model1) # Stepwise model selection
```

```
## Start:  AIC=519.73
## y ~ chol + hdl + ratio + age + gender + height + weight + frame +
##     bp.1s + bp.1d + waist + hip
##
##          Df Sum of Sq    RSS    AIC
## - weight  1     0.001 1409.0 517.73
## - hip     1     0.026 1409.1 517.73
## - frame   1     0.297 1409.3 517.80
## - hdl     1     0.670 1409.7 517.90
## - chol    1     1.356 1410.4 518.08
## - gender  1     1.602 1410.7 518.14
## - bp.1d   1     2.508 1411.5 518.38
## - bp.1s   1     3.281 1412.3 518.58
## - waist   1     5.733 1414.8 519.22
## - height  1     6.772 1415.8 519.49
## <none>                1409.0 519.73
## - ratio   1    20.044 1429.1 522.91
## - age     1    64.328 1473.4 534.11
##
## Step:  AIC=517.73
## y ~ chol + hdl + ratio + age + gender + height + frame + bp.1s +
##     bp.1d + waist + hip
##
##          Df Sum of Sq    RSS    AIC
## - hip     1     0.052 1409.1 515.74
## - frame   1     0.312 1409.4 515.81
## - hdl     1     0.669 1409.7 515.90
## - chol    1     1.358 1410.4 516.08
## - gender  1     1.688 1410.7 516.17
## - bp.1d   1     2.541 1411.6 516.39
## - bp.1s   1     3.332 1412.4 516.59
## - waist   1     7.111 1416.2 517.57
## - height  1     7.638 1416.7 517.71
## <none>                1409.0 517.73
## - ratio   1    20.076 1429.1 520.92
## - age     1    68.578 1477.6 533.17
##
## Step:  AIC=515.74
## y ~ chol + hdl + ratio + age + gender + height + frame + bp.1s +
##     bp.1d + waist
##
##          Df Sum of Sq    RSS    AIC
## - frame   1     0.332 1409.4 513.83
## - hdl     1     0.722 1409.8 513.93
## - chol    1     1.322 1410.4 514.09
## - gender  1     1.682 1410.8 514.18
## - bp.1d   1     2.567 1411.7 514.41
## - bp.1s   1     3.309 1412.4 514.60
## - height  1     7.613 1416.7 515.72
## <none>                1409.1 515.74
## - waist   1    17.403 1426.5 518.25
## - ratio   1    20.597 1429.7 519.07
## - age     1    71.994 1481.1 532.03
##
## Step:  AIC=513.83
## y ~ chol + hdl + ratio + age + gender + height + bp.1s + bp.1d +
##     waist
##
##          Df Sum of Sq    RSS    AIC
## - hdl     1     0.805 1410.2 512.04
## - chol    1     1.277 1410.7 512.16
## - gender  1     2.303 1411.7 512.43
## - bp.1d   1     2.502 1411.9 512.48
## - bp.1s   1     3.232 1412.7 512.67
## <none>                1409.4 513.83
## - height  1     8.376 1417.8 514.00
## - waist   1    19.367 1428.8 516.84
## - ratio   1    20.778 1430.2 517.20
## - age     1    71.733 1481.2 530.05
##
## Step:  AIC=512.04
## y ~ chol + ratio + age + gender + height + bp.1s + bp.1d + waist
##
##          Df Sum of Sq    RSS    AIC
## - bp.1d   1     2.443 1412.7 510.67
## - gender  1     2.520 1412.8 510.69
## - bp.1s   1     3.230 1413.5 510.88
## <none>                1410.2 512.04
## - chol    1     8.943 1419.2 512.36
## - height  1     9.218 1419.5 512.43
## - waist   1    18.613 1428.8 514.85
## - ratio   1    65.821 1476.1 526.78
## - age     1    73.102 1483.3 528.58
##
## Step:  AIC=510.67
## y ~ chol + ratio + age + gender + height + bp.1s + waist
##
##          Df Sum of Sq    RSS    AIC
## - bp.1s   1     1.067 1413.7 508.95
## - gender  1     2.838 1415.5 509.41
## <none>                1412.7 510.67
## - chol    1     7.876 1420.5 510.71
## - height  1     9.092 1421.8 511.03
## - waist   1    17.353 1430.0 513.15
## - ratio   1    68.835 1481.5 526.13
## - age     1    90.379 1503.1 531.43
##
## Step:  AIC=508.95
## y ~ chol + ratio + age + gender + height + waist
##
##          Df Sum of Sq    RSS    AIC
## - gender  1     2.760 1416.5 507.66
## <none>                1413.7 508.95
## - chol    1     8.656 1422.4 509.19
## - height  1     8.989 1422.7 509.28
## - waist   1    19.182 1432.9 511.90
## - ratio   1    68.037 1481.8 524.20
## - age     1   114.376 1528.1 535.50
##
## Step:  AIC=507.66
## y ~ chol + ratio + age + height + waist
```

```
##
##          Df Sum of Sq    RSS    AIC
## - height  1     6.614 1423.1 507.37
## <none>                1416.5 507.66
## - chol    1     9.419 1425.9 508.10
## - waist   1    21.885 1438.4 511.29
## - ratio   1    66.048 1482.5 522.39
## - age     1   111.900 1528.4 533.57
##
## Step:  AIC=507.37
## y ~ chol + ratio + age + waist
##
##         Df Sum of Sq    RSS    AIC
## <none>               1423.1 507.37
## - chol   1     8.094 1431.2 507.46
## - waist  1    22.682 1445.8 511.18
## - ratio  1    71.511 1494.6 523.37
## - age    1   107.713 1530.8 532.15
```

```
# Summarize results
summary(Model1)
```

```
##
## Call:
## lm(formula = y ~ ., data = X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1026 -1.1404 -0.4033  0.3779  9.6117
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.6855419  3.1624253  -1.165   0.2446
## chol         0.0026899  0.0046078   0.584   0.5597
## hdl          0.0058946  0.0143623   0.410   0.6817
## ratio        0.3510530  0.1564389   2.244   0.0254 *
## age          0.0331220  0.0082390   4.020 7.11e-05 ***
## gendermale  -0.2160856  0.3406172  -0.634   0.5262
## height       0.0544762  0.0417640   1.304   0.1930
## weight      -0.0001388  0.0072834  -0.019   0.9848
## frame       -0.0489722  0.1793965  -0.273   0.7850
## bp.1s        0.0061097  0.0067299   0.908   0.3646
## bp.1d       -0.0082551  0.0103996  -0.794   0.4278
## waist        0.0507672  0.0423030   1.200   0.2309
## hip         -0.0038590  0.0478056  -0.081   0.9357
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.995 on 354 degrees of freedom
## Multiple R-squared:  0.226,  Adjusted R-squared:  0.1998
## F-statistic: 8.614 on 12 and 354 DF,  p-value: 2.076e-14
```

```
summary(Model2)
```

```
##
## Call:
## lm(formula = y ~ chol + ratio + age + waist, data = X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2007 -1.1169 -0.4211  0.4178  9.5154
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.076917   0.811062   0.095   0.9245
## chol        0.003930   0.002739   1.435   0.1522
## ratio       0.299880   0.070312   4.265 2.55e-05 ***
## age         0.034698   0.006629   5.234 2.81e-07 ***
## waist       0.045601   0.018984   2.402   0.0168 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.983 on 362 degrees of freedom
## Multiple R-squared:  0.2183, Adjusted R-squared:  0.2096
## F-statistic: 25.27 on 4 and 362 DF,  p-value: < 2.2e-16
```

```
complete_case = summary(Model2)
```

Step 7: Imputation Methods

A. Mean Imputation

```
#simple model, imputation with mean
M.imp = mice(DM,method = "mean",m=20)
```

```
## Warning: Number of logged events: 1
```

```
names(M.imp)
M.imp$imp
```

```
y=DM[,4]
X=DM[,-4]
Model3=with(M.imp,lm(y~ratio+age+waist,data=X))
```

```
#Model3=step(Model3)
# Run the pooled model summary
pool_summary <- summary(pool(Model3))

# Extract the coefficient estimates and standard errors
mean_coef <- round(pool_summary[, c("estimate", "std.error")], 2)
mean_imp_results = summary(pool(Model3))

# Access estimates and standard errors
mean_coef[, 1] # estimates
```

```
## [1] 0.79 0.32 0.04 0.04
```

```
mean_coef[, 2] # standard errors
```

```
## [1] 0.71 0.06 0.01 0.02
```

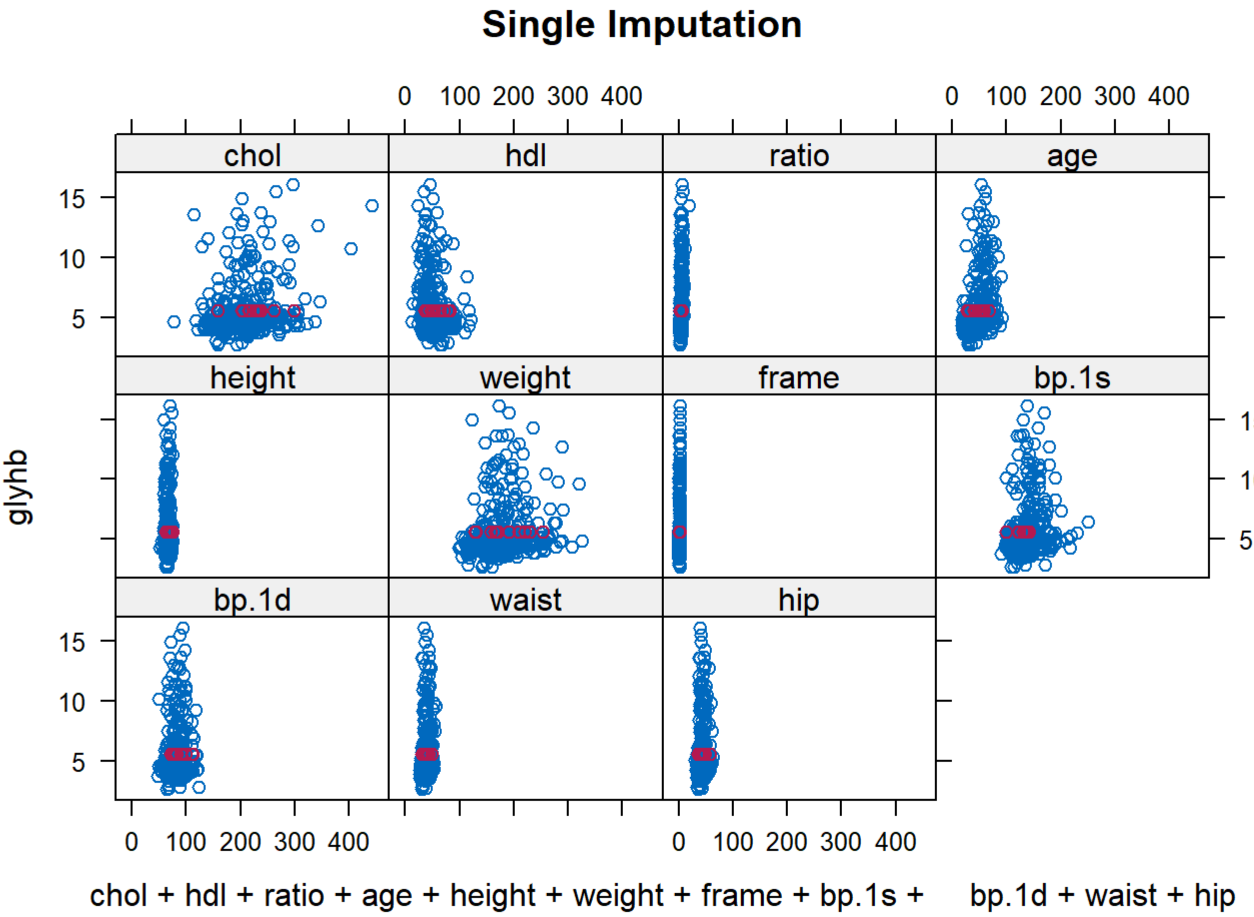
```
# Check structure of the mean_imp_results data frame to identify numeric columns
str(mean_imp_results)
```

```
## Classes 'mipo.summary' and 'data.frame': 4 obs. of  6 variables:
##  $ term     : Factor w/ 4 levels "(Intercept)",..: 1 2 3 4
##  $ estimate : num  0.7915 0.319 0.0384 0.0414
##  $ std.error: num  0.71291 0.06258 0.00635 0.0189
##  $ statistic: num  1.11 5.1 6.05 2.19
##  $ df       : num  381 381 381 381
##  $ p.value  : num  2.68e-01 5.42e-07 3.56e-09 2.92e-02
```

```
# Extract only the numeric columns for rounding
numeric_columns <- sapply(mean_imp_results, is.numeric)

# Round the numeric columns to 3 decimal places
mean_coef <- mean_imp_results
mean_coef[, numeric_columns] <- round(mean_imp_results[, numeric_columns], 3)

# View the rounded results
mean_coef
```

```
##          term estimate std.error statistic      df p.value
## 1 (Intercept)    0.791     0.713     1.110 380.977   0.268
## 2       ratio    0.319     0.063     5.098 380.977   0.000
## 3         age    0.038     0.006     6.045 380.977   0.000
## 4       waist    0.041     0.019     2.188 380.977   0.029
```

```
xyplot(M.imp,glyhb~chol+hdl+ratio+age+height
       +weight+frame+bp.1s+bp.1d+waist+hip,
       main="Single Imputation")
```



B. Multiple Imputation Using MICE

```
# Multiple imputation using MICE
C.imp = mice(DM, m = 20, method = "pmm") # Predictive Mean Matching
```

```
## 
##  iter imp variable
##   1   1  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   2  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   3  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   4  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   5  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   6  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   7  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   8  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   9  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   10  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   11  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   12  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   13  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   14  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   15  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   16  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   17  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   18  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   19  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   1   20  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   1  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   2  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   3  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   4  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   5  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   6  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   7  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   8  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   9  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   10  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   11  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   12  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   13  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   14  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   15  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   16  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   17  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   18  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   19  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   2   20  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   1  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   2  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   3  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   4  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   5  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   6  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   7  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   8  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   9  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   10  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   11  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   12  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   13  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   14  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   15  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   16  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   17  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   18  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   19  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   3   20  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   1  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   2  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   3  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   4  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   5  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   6  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   7  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   8  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   9  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   10  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   11  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   12  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   13  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   14  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   15  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   16  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   17  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   18  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   19  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   4   20  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   1  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   2  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   3  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   4  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   5  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   6  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   7  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   8  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   9  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   10  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   11  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   12  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   13  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   14  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   15  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   16  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   17  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   18  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   19  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
##   5   20  chol  hdl  ratio  glyhb  height  weight  frame  bp.1s  bp.1d  waist  hip
```

```
## Warning: Number of logged events: 1
```

```
# Fit models on imputed datasets
model = with(data = C.imp, exp = lm(glyhb ~ chol + hdl + ratio + age + height + weight + fram
e + bp.1s + bp.1d + waist + hip))
multiple_imp_results = summary(pool(model))

# Model refinement (sequential reduction)
model8 = with(data = C.imp, exp = lm(glyhb ~ ratio + age + waist))
model8_results = summary(pool(model8))
```

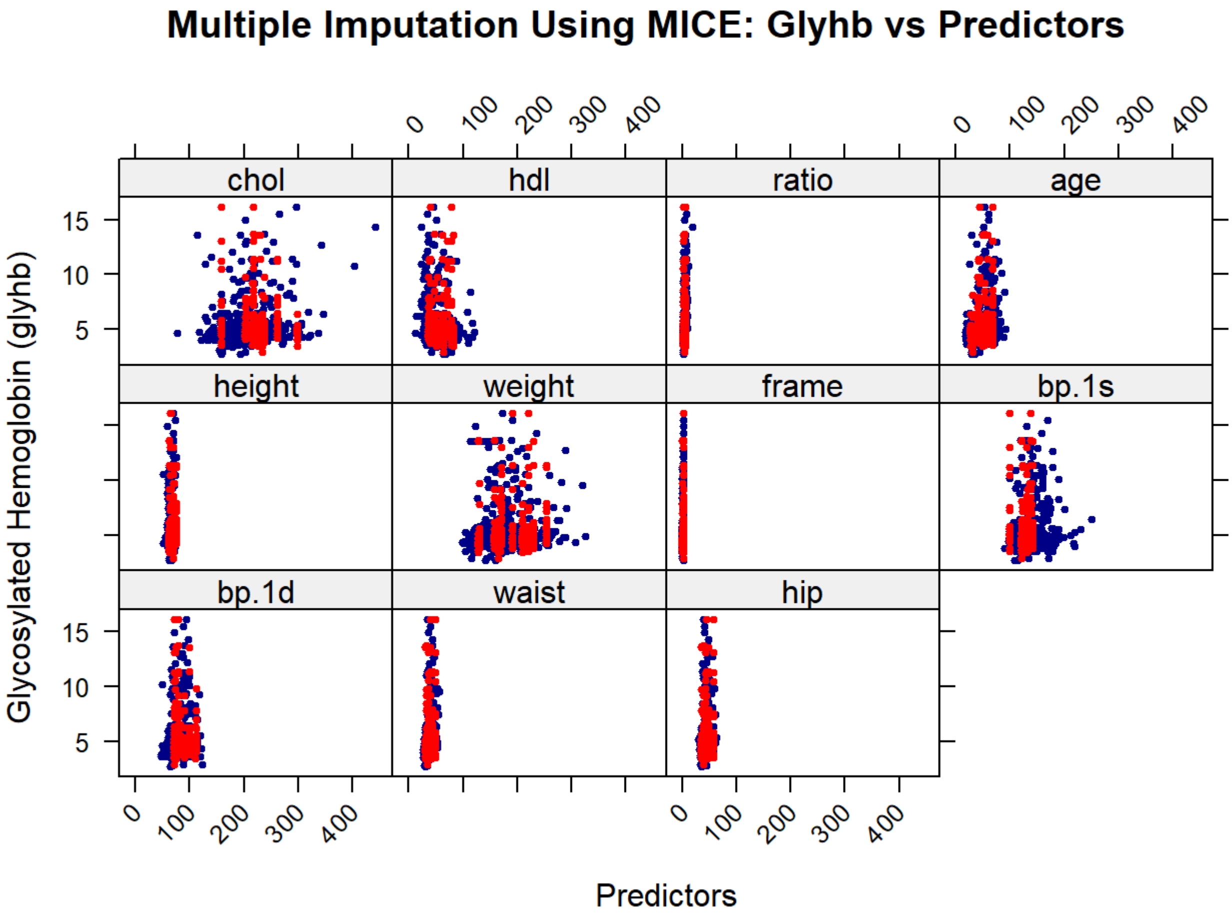
```
# Check the structure of the multiple_imp_results data frame to identify numeric columns
str(multiple_imp_results)
```

```
## Classes 'mipo.summary' and 'data.frame': 12 obs. of  6 variables:
##  $ term     : Factor w/ 12 levels "(Intercept)",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ estimate : num  -2.96664 0.00114 0.0094 0.35985 0.03548 ...
##  $ std.error: num  2.96308 0.00458 0.01431 0.15685 0.00815 ...
##  $ statistic: num  -1.001 0.249 0.657 2.294 4.351 ...
##  $ df       : num  331 333 328 361 340 ...
##  $ p.value  : num  0.31746 0.8032 0.511715 0.02235 0.000018 ...
```

```
# Identify numeric columns (i.e., exclude non-numeric columns like 'term')
numeric_columns <- sapply(multiple_imp_results, is.numeric)

# Round only the numeric columns to 3 decimal places
multiple_coef <- multiple_imp_results
multiple_coef[, numeric_columns] <- round(multiple_imp_results[, numeric_columns], 3)

# View the rounded results
multiple_coef
```

```
##           term estimate std.error statistic      df p.value
## 1  (Intercept)   -2.967     2.963    -1.001 331.041   0.317
## 2         chol    0.001     0.005     0.249 333.206   0.803
## 3          hdl    0.009     0.014     0.657 328.406   0.512
## 4        ratio    0.360     0.157     2.294 361.391   0.022
## 5          age    0.035     0.008     4.351 340.387   0.000
## 6       height    0.041     0.034     1.211 338.273   0.227
## 7       weight    0.000     0.007    -0.066 292.700   0.947
## 8        frame   -0.023     0.172    -0.135 361.139   0.893
## 9        bp.1s    0.006     0.007     0.889 335.257   0.375
## 10       bp.1d   -0.011     0.010    -1.103 370.106   0.271
## 11       waist    0.034     0.042     0.803 358.796   0.423
## 12         hip    0.019     0.044     0.420 343.146   0.675
```

```
# xyplot for Multiple Imputation Using MICE
xyplot(C.imp, glyhb ~ chol + hdl + ratio + age + height + weight + frame + bp.1s + bp.1d + wa
ist + hip,
       main = "Multiple Imputation Using MICE: Glyhb vs Predictors",
       pch = 20,
       cex = 0.7,
       col = c("darkblue", "red"),
       xlab = "Predictors",
       ylab = "Glycosylated Hemoglobin (glyhb)",
       scales = list(x = list(rot = 45)))  # Rotate x-axis labels for better visibility
```



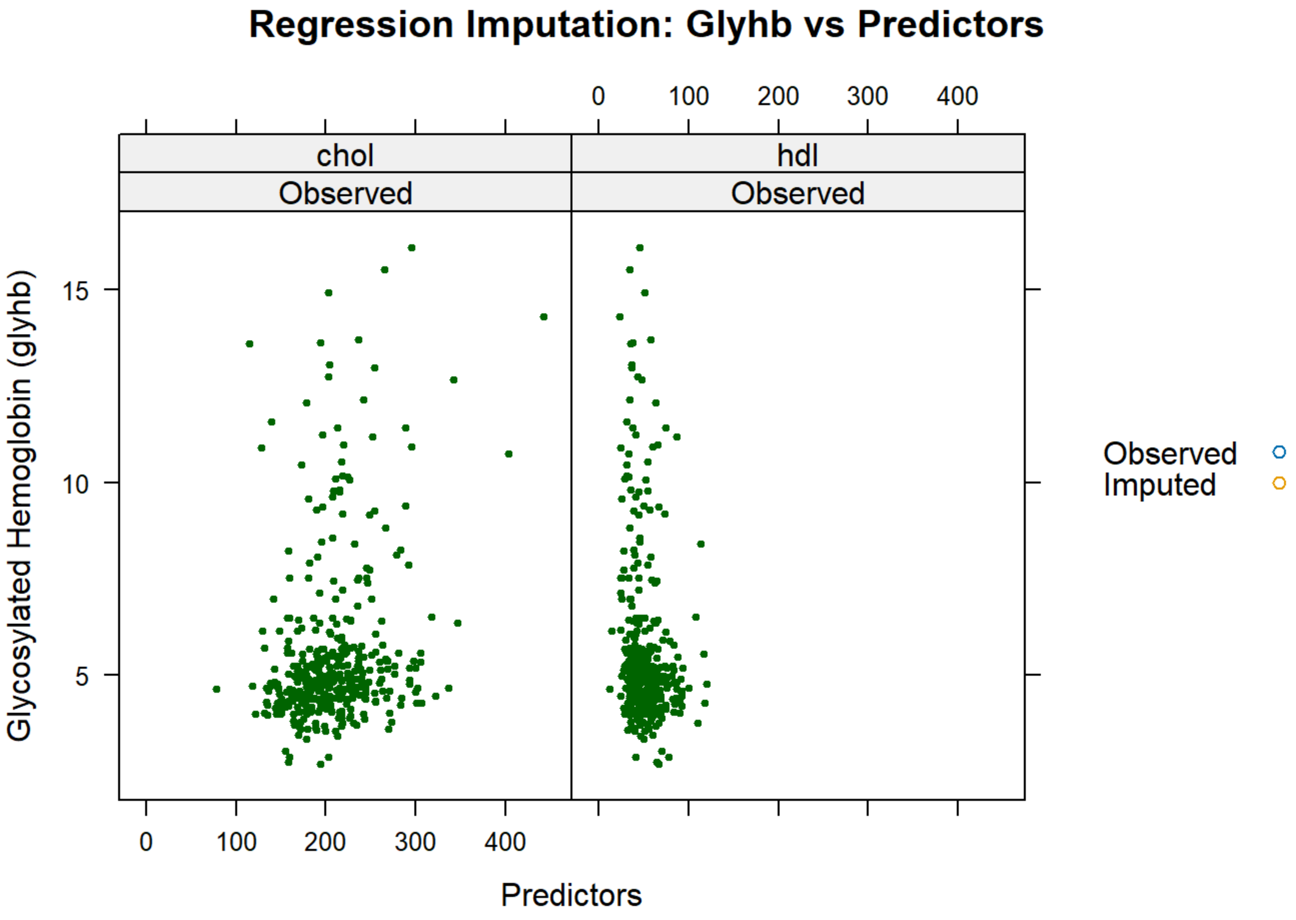
C. Regression Imputation

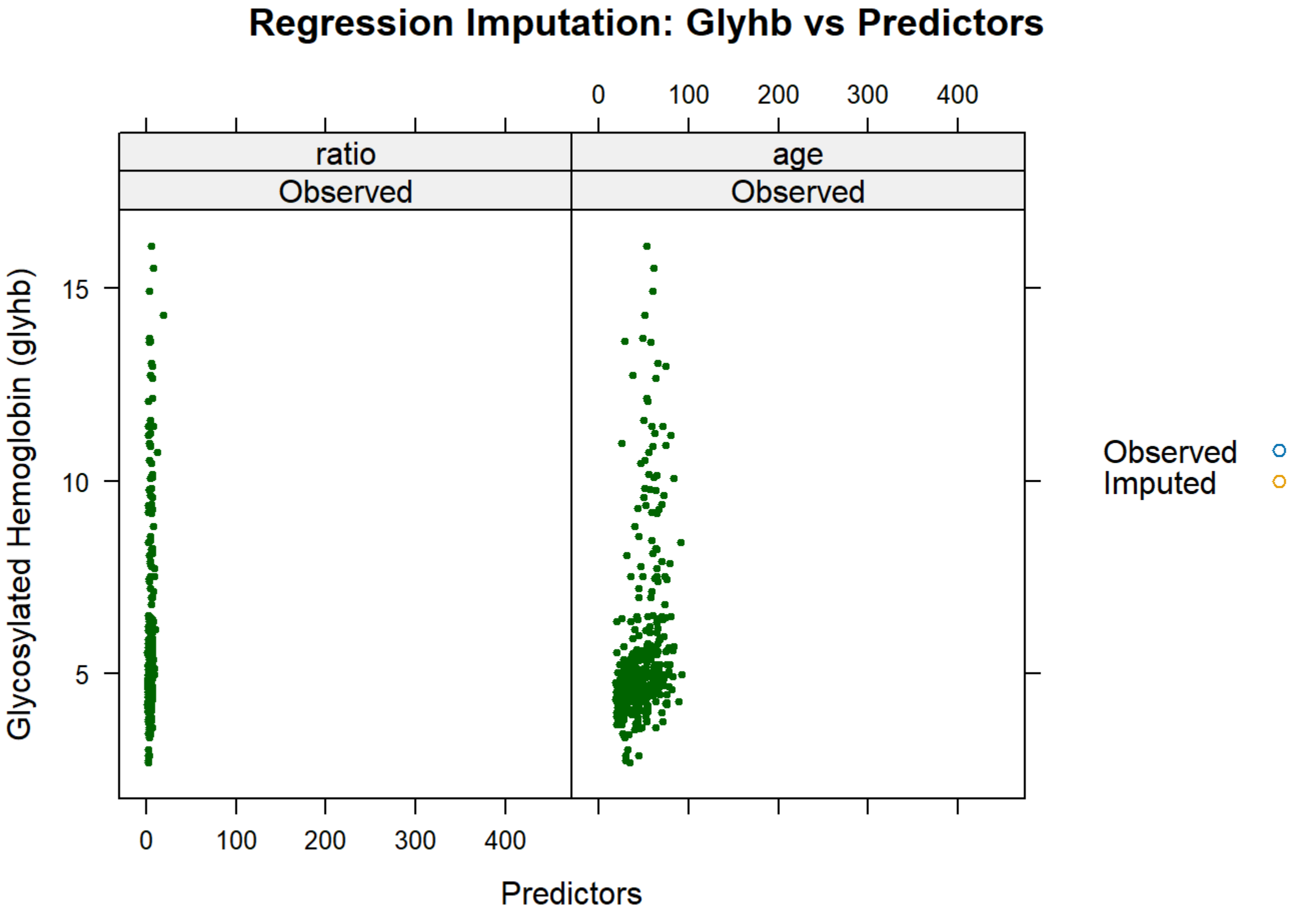
```
# Fit a regression model for prediction
reg_model = lm(glyhb ~ ratio + age + waist + bp.1s + bp.1d, data = DM, na.action = na.omit)

# Predict missing values and update the dataset
predicted_values = predict(reg_model, newdata = DM[is.na(DM$glyhb), ])
DM$glyhb[is.na(DM$glyhb)] = predicted_values
```

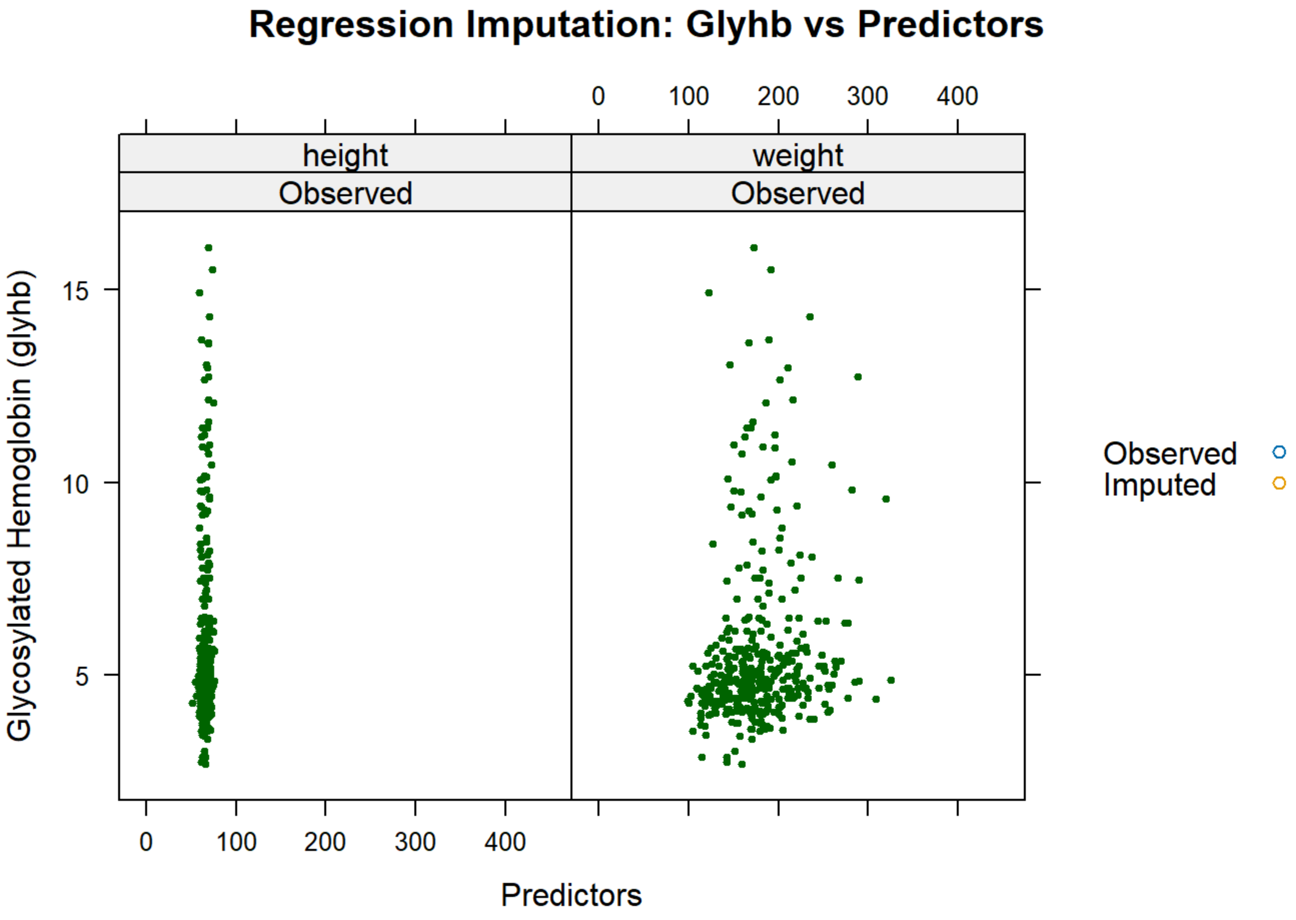
```
DM$source <- ifelse(is.na(DM$glyhb), "Imputed", "Observed")  # Tag observed vs. imputed
DM$source <- factor(DM$source, levels = c("Observed", "Imputed"))

# Plot for Regression Imputation
library(lattice)
```

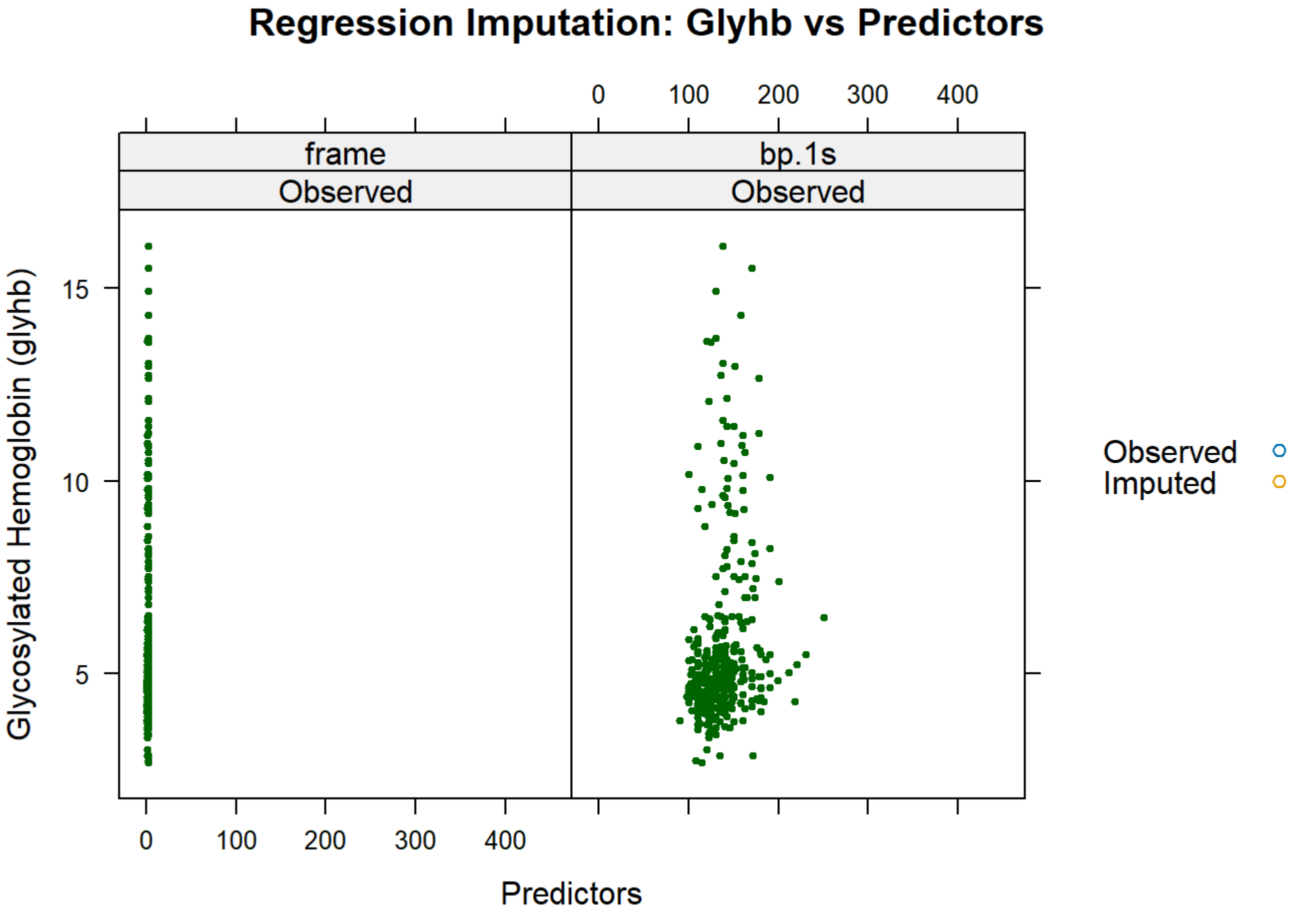
```
## Warning: package 'lattice' was built under R version 4.3.3
```

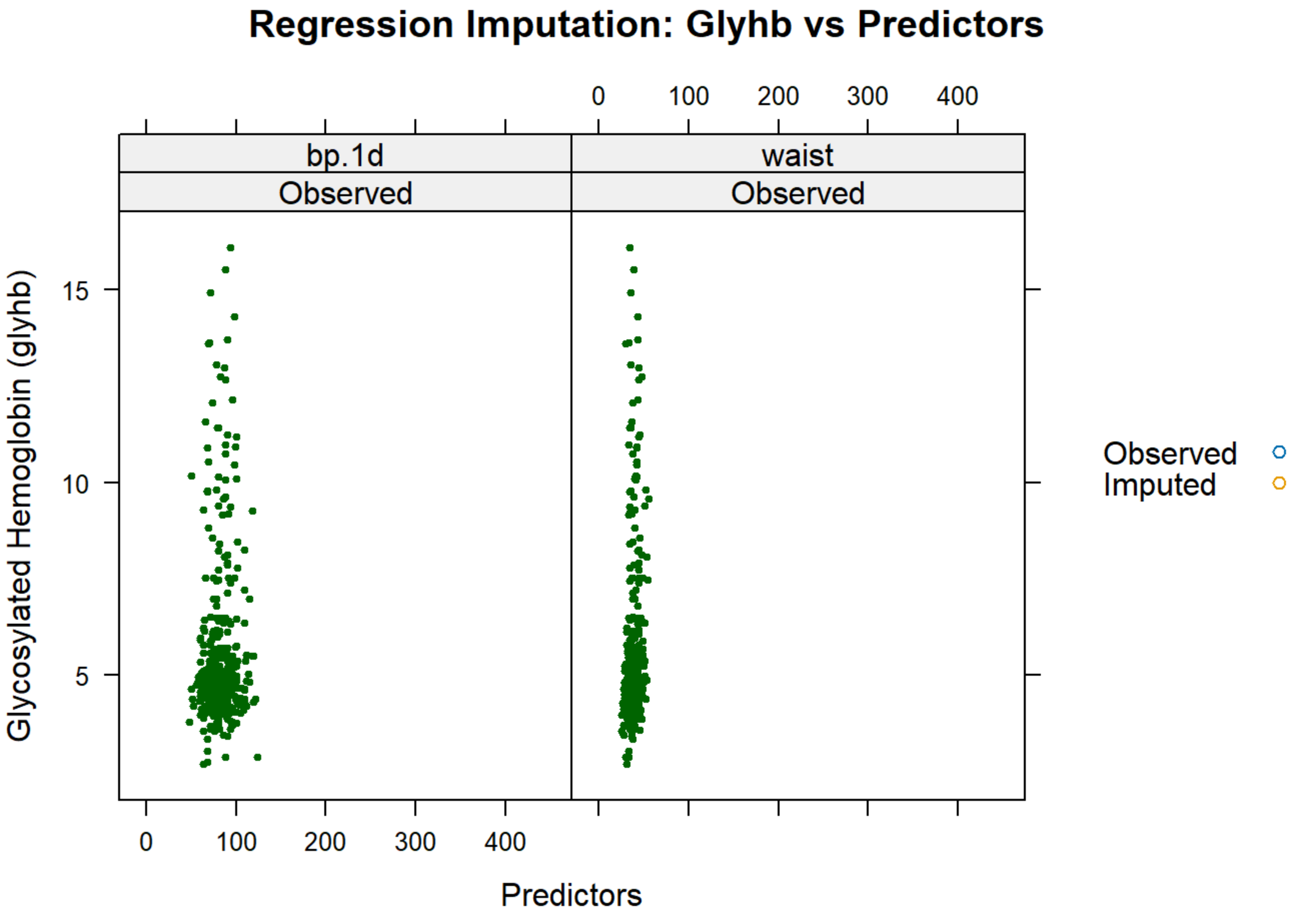
```
xyplot(glyhb ~ chol + hdl + ratio + age + height + weight + frame + bp.1s + bp.1d + waist + h
ip | source,
       data = DM,
       groups = source,
       auto.key = list(space = "right", points = TRUE, lines = FALSE),
       pch = 20,
       cex = 0.7,
       col = c("darkgreen", "orange"),
       main = "Regression Imputation: Glyhb vs Predictors",
       xlab = "Predictors",
       ylab = "Glycosylated Hemoglobin (glyhb)",
       layout = c(2, 1))  # Layout for Observed and Imputed
```

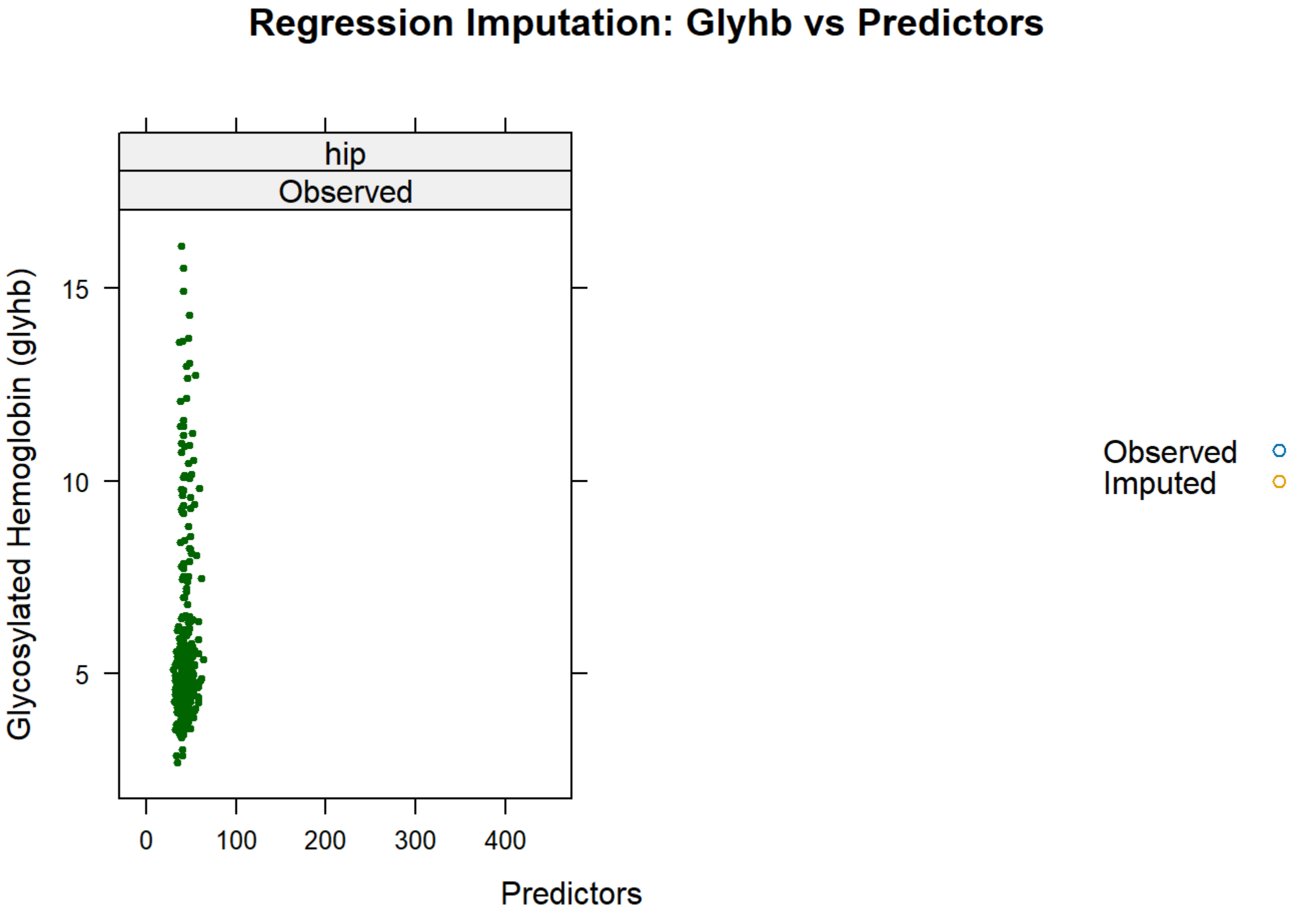

Regression Imputation: Glyhb vs Predictors
chol
Observed
hdl
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed ○
Imputed ○


Regression Imputation: Glyhb vs Predictors
ratio
Observed
age
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed ○
Imputed ○


Regression Imputation: Glyhb vs Predictors
height
Observed
weight
Observed
Glycosylated Hemoglobin (glyhb)
5
10
15
0
100
200
300
400
Predictors
Observed ○
Imputed ○


Regression Imputation: Glyhb vs Predictors
frame
Observed
bp.1s
Observed
Glycosylated Hemoglobin (glyhb)
5
10
15
0
100
200
300
400
Predictors
Observed ○
Imputed ○


Regression Imputation: Glyhb vs Predictors
bp.1d
Observed
waist
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed
Imputed

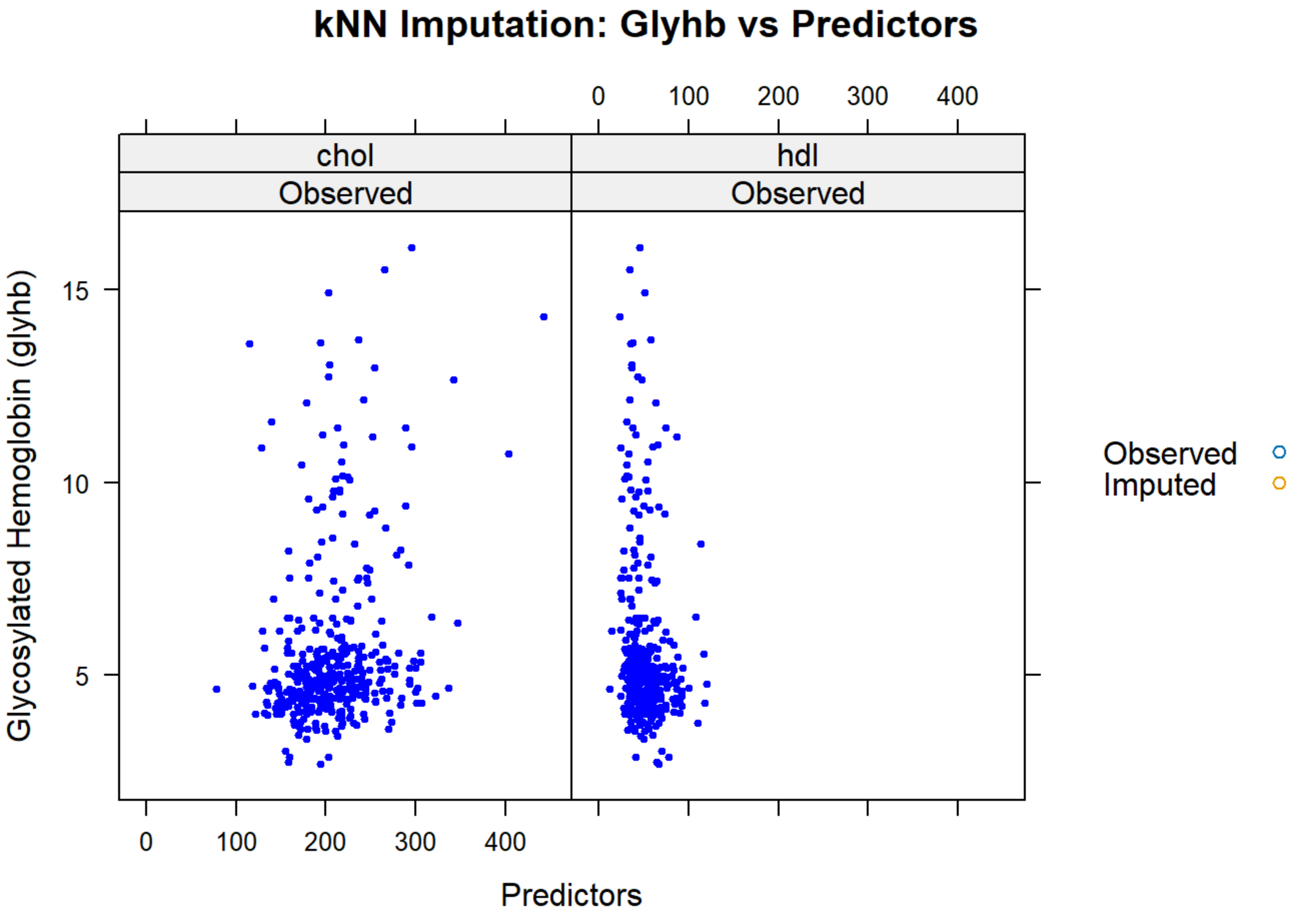

Regression Imputation: Glyhb vs Predictors
hip
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed
Imputed

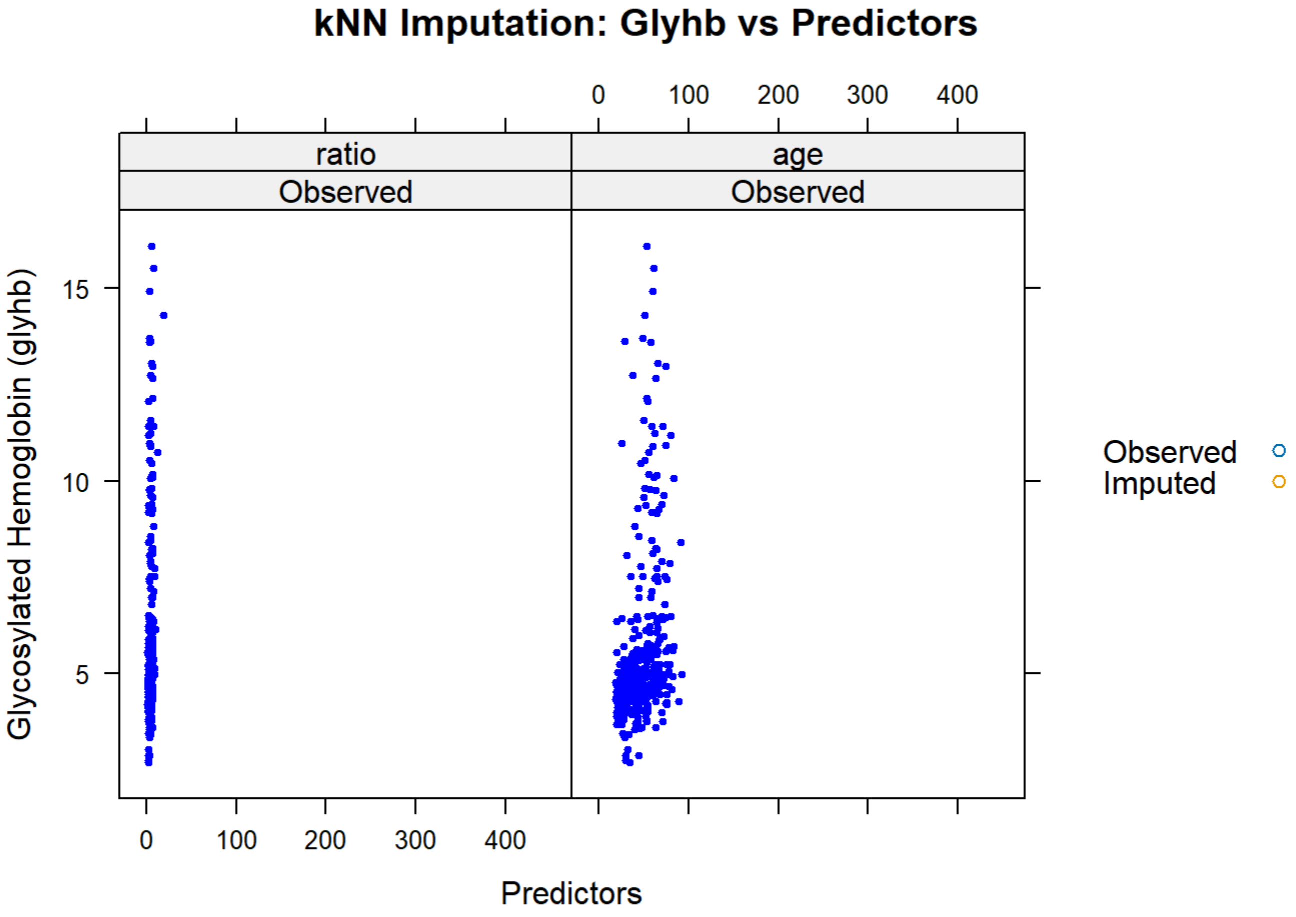
D. k-Nearest Neighbor (kNN) Imputation

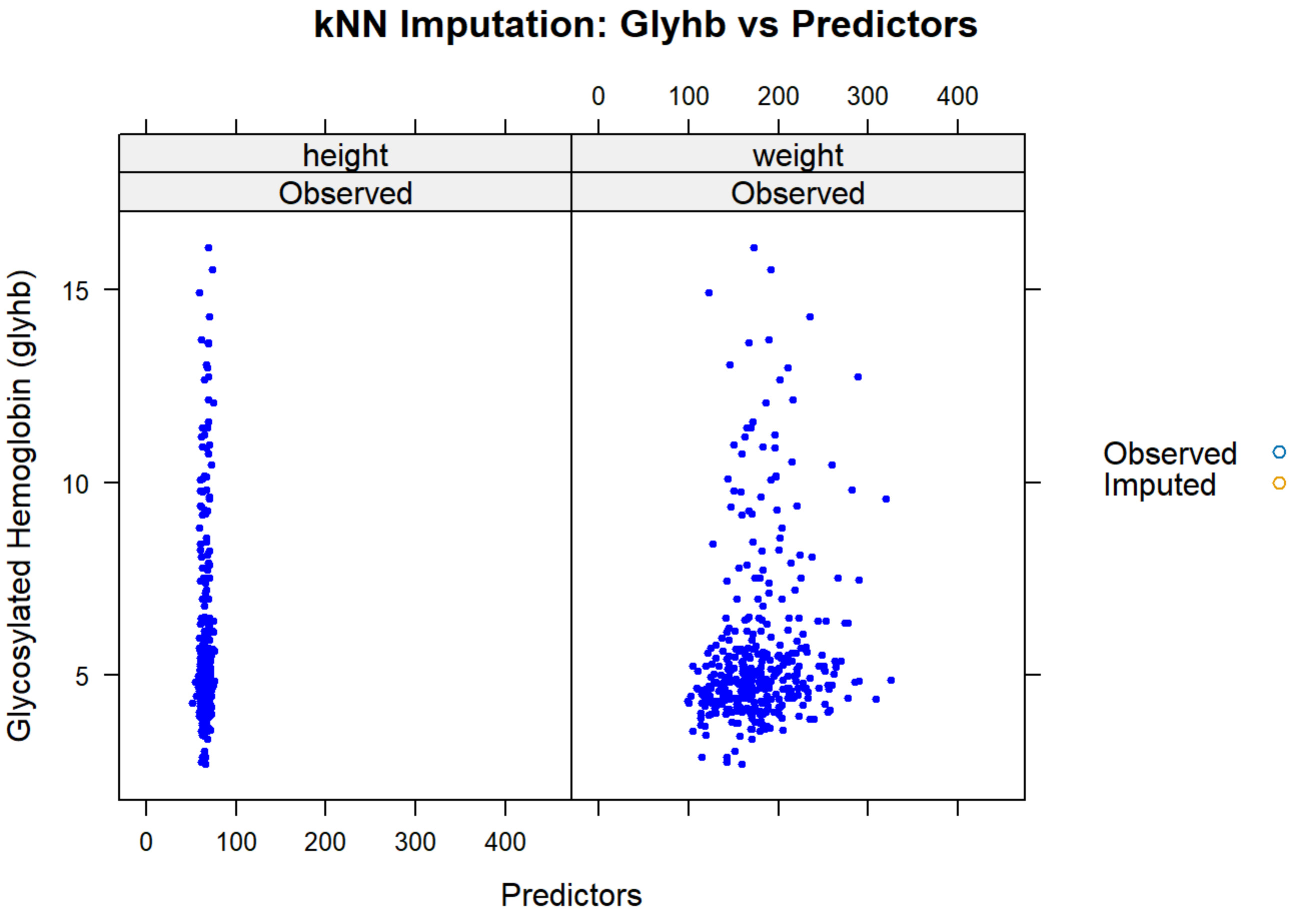
```
# Perform kNN imputation
library(VIM)
DM_knn = kNN(DM, k = 5, variable = "glyhb")

# Check imputed values
head(DM_knn)
```

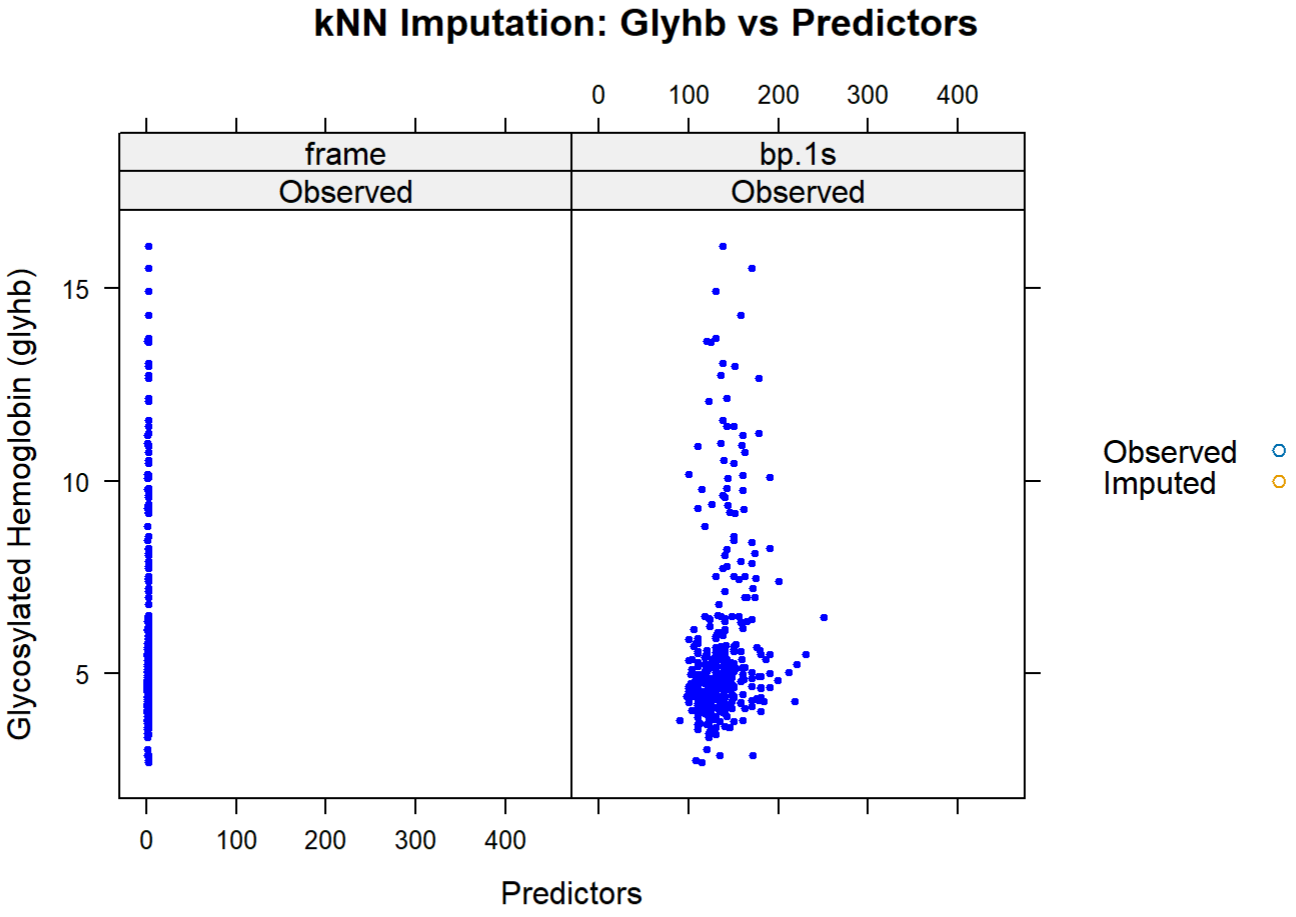
```
##   chol hdl ratio glyhb age gender height weight frame bp.1s bp.1d waist hip
## 1  203  56   3.6  4.31  46 female     62    121     2   118    59    29  38
## 2  165  24   6.9  4.44  29 female     64    218     3   112    68    46  48
## 3  228  37   6.2  4.64  58 female     61    256     3   190    92    49  57
## 4   78  12   6.5  4.63  67   male     67    119     3   110    50    33  38
## 5  249  28   8.9  7.72  64   male     68    183     2   138    80    44  41
## 6  248  69   3.6  4.81  34   male     71    190     3   132    86    36  42
##     source glyhb_imp
## 1 Observed     FALSE
## 2 Observed     FALSE
## 3 Observed     FALSE
## 4 Observed     FALSE
## 5 Observed     FALSE
## 6 Observed     FALSE
```

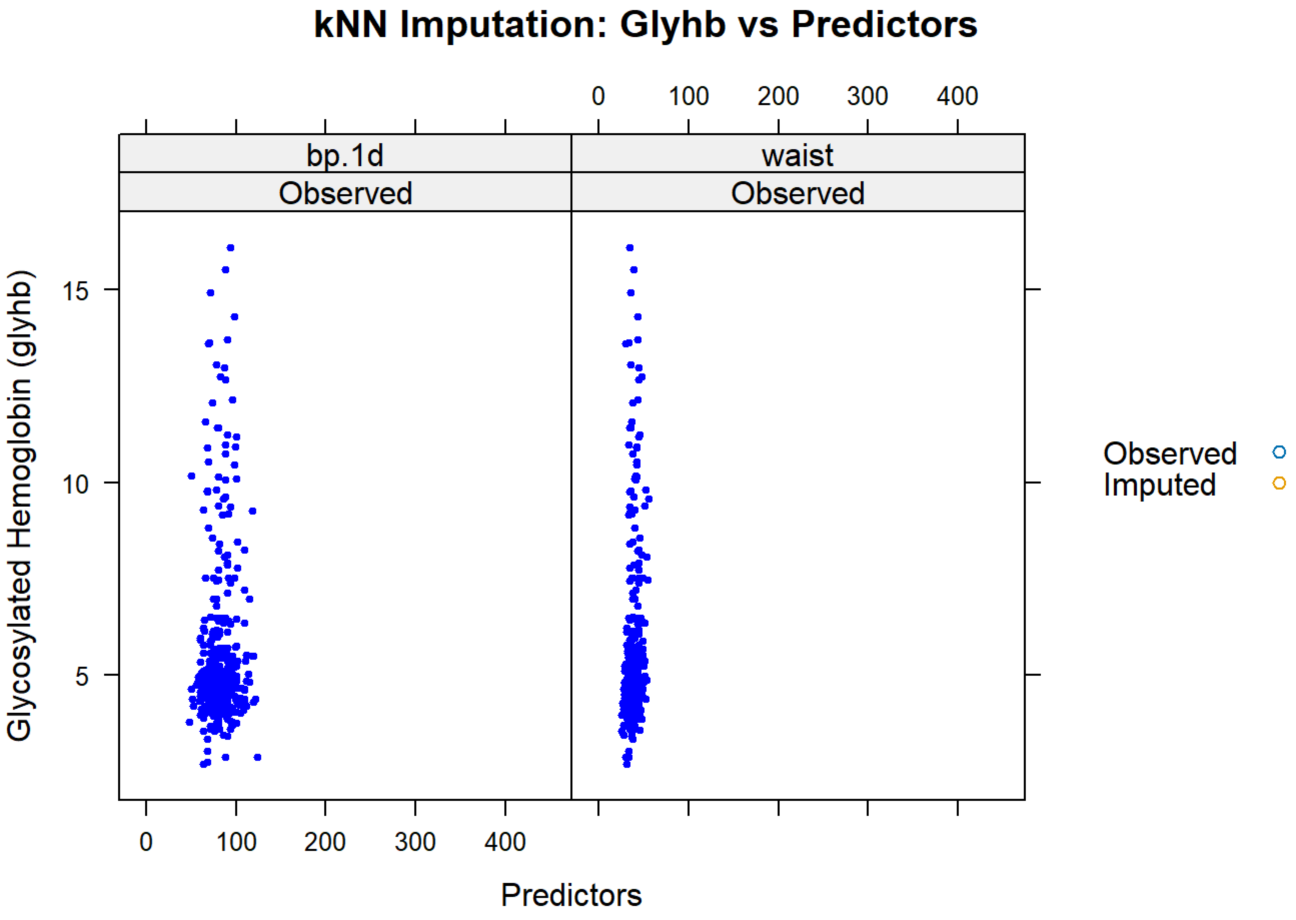
```
DM$source <- ifelse(is.na(DM$glyhb), "Imputed", "Observed")  # Tag observed vs. imputed
DM$source <- factor(DM$source, levels = c("Observed", "Imputed"))

# Plot for kNN Imputation
xyplot(glyhb ~ chol + hdl + ratio + age + height + weight + frame + bp.1s + bp.1d + waist + h
ip | source,
       data = DM,
       groups = source,
       auto.key = list(space = "right", points = TRUE, lines = FALSE),
       pch = 20,
       cex = 0.7,
       col = c("blue", "red"),
       main = "kNN Imputation: Glyhb vs Predictors",
       xlab = "Predictors",
       ylab = "Glycosylated Hemoglobin (glyhb)",
       layout = c(2, 1))  # Layout for Observed and Imputed
```

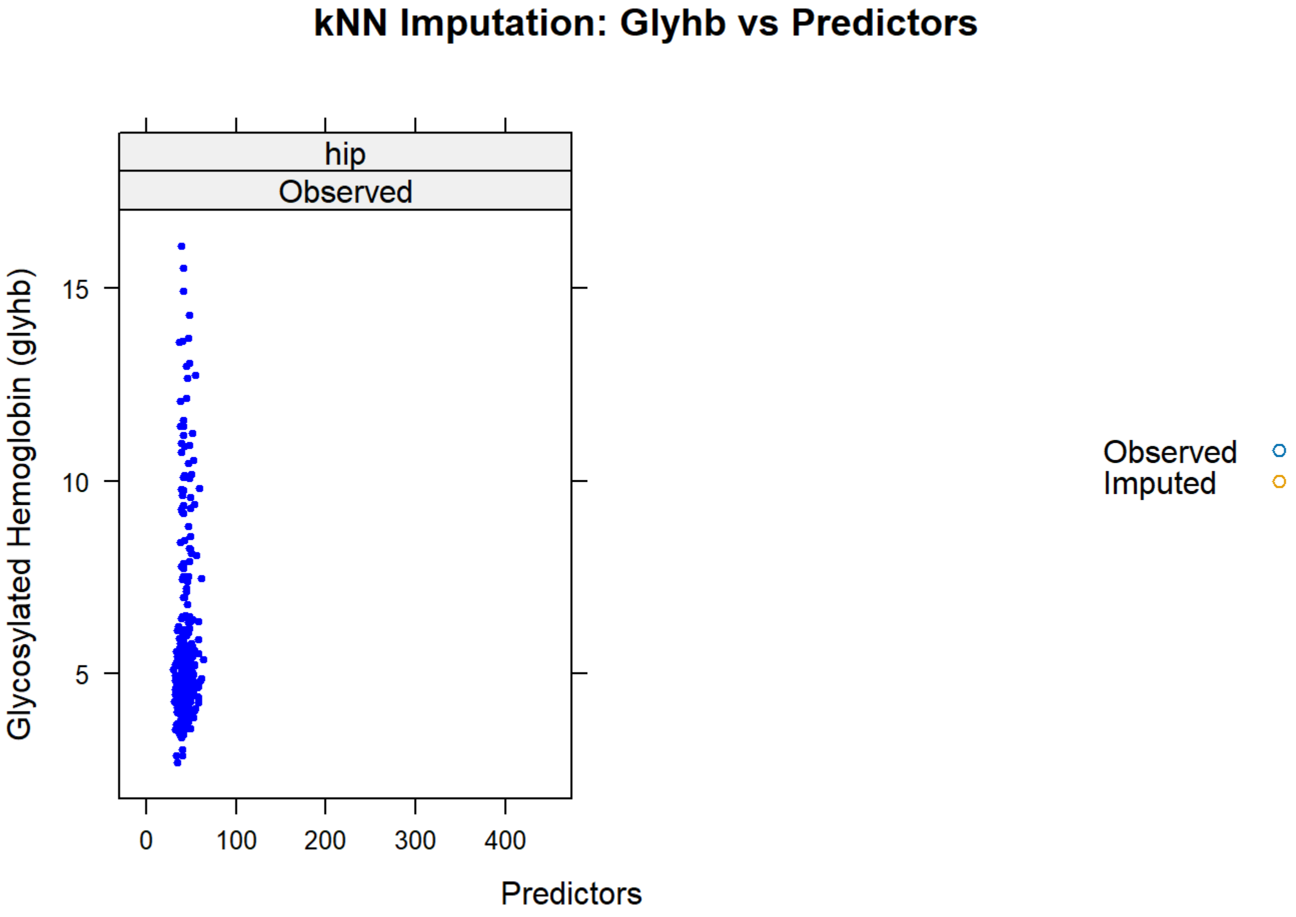

kNN Imputation: Glyhb vs Predictors
chol
Observed
hdl
Observed
Glycosylated Hemoglobin (glyhb)
5
10
15
0
100
200
300
400
Predictors
Observed ○
Imputed ○


kNN Imputation: Glyhb vs Predictors
ratio
Observed
age
Observed
Glycosylated Hemoglobin (glyhb)
5
10
15
0
100
200
300
400
Predictors
Observed ○
Imputed ○


kNN Imputation: Glyhb vs Predictors
height
Observed
weight
Observed
Glycosylated Hemoglobin (glyhb)
15
10
5
0
100
200
300
400
Predictors
Observed ○
Imputed ○

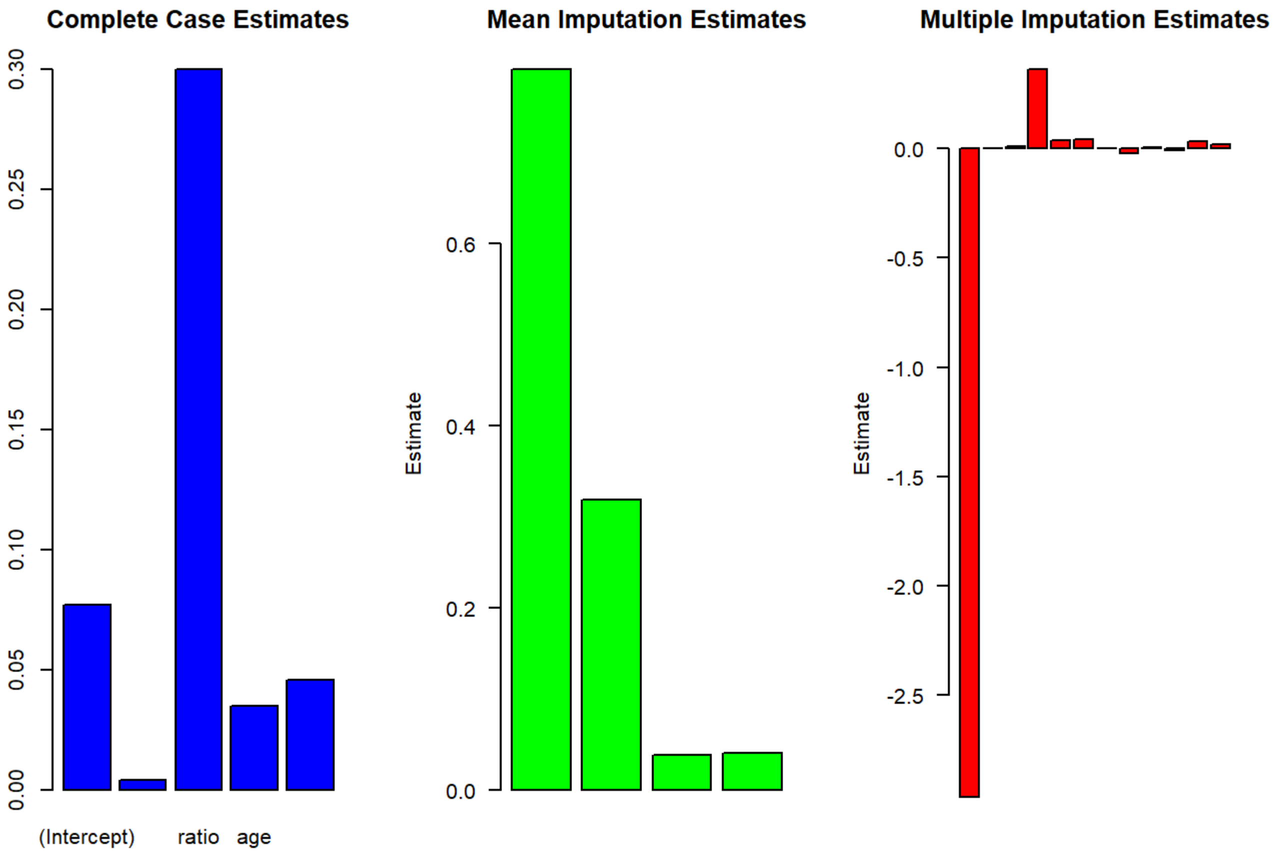

kNN Imputation: Glyhb vs Predictors
frame
Observed
bp.1s
Observed
Glycosylated Hemoglobin (glyhb)
15
10
5
0
100
200
300
400
Predictors
Observed ○
Imputed ○


kNN Imputation: Glyhb vs Predictors
bp.1d
Observed
waist
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed ○
Imputed ○

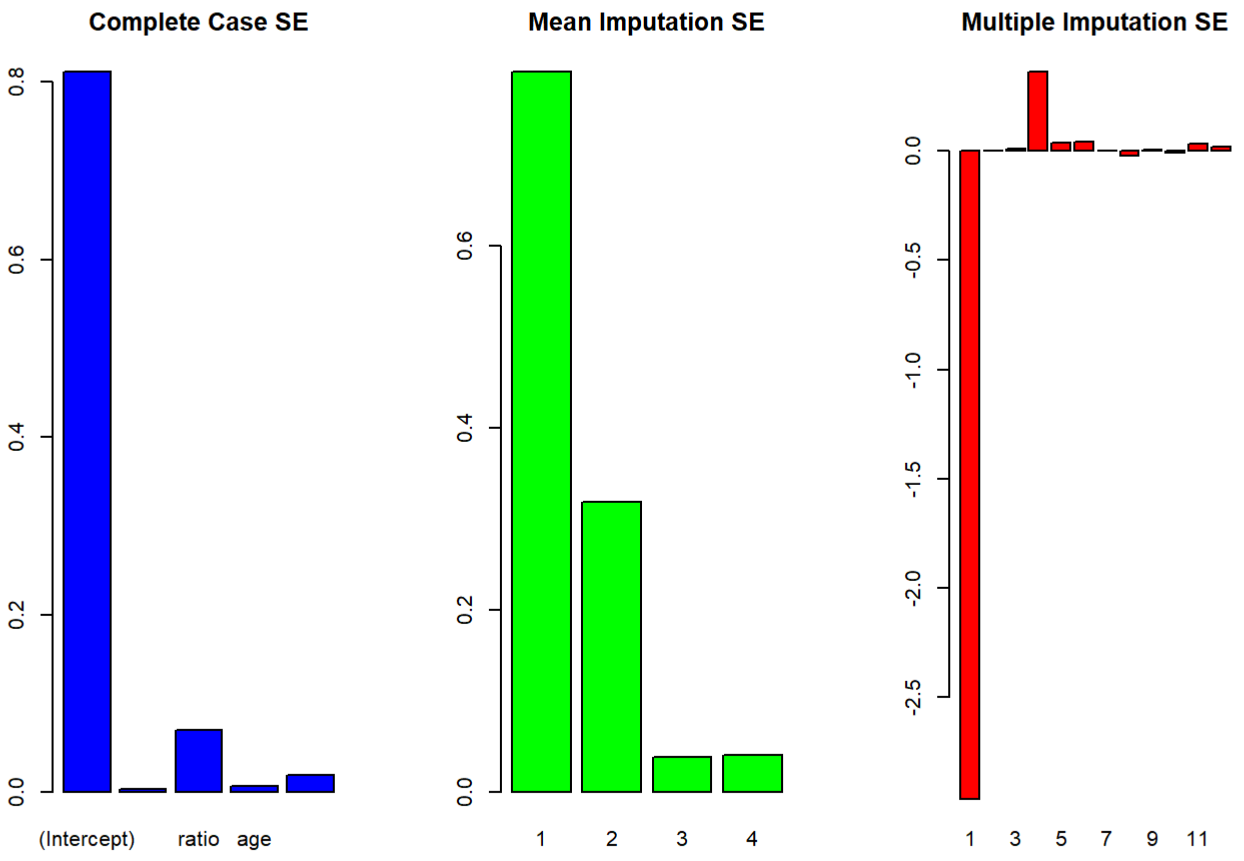

kNN Imputation: Glyhb vs Predictors
hip
Observed
Glycosylated Hemoglobin (glyhb)
Predictors
0
100
200
300
400
5
10
15
Observed ○
Imputed ○

Step 7: Model Comaprison

```
# Extract coefficients and standard errors
complete_case_coef = round(complete_case$coefficients, 3)
mean_coef[, numeric_columns] <- round(mean_imp_results[, numeric_columns], 3)
multiple_coef[, numeric_columns] <- round(multiple_imp_results[, numeric_columns], 3)

# Visualize comparisons
par(mfrow = c(1, 3))

# Coefficients comparison
barplot(complete_case_coef[,1], col = "blue", names.arg = rownames(complete_case_coef), main
= "Complete Case Estimates")

barplot(mean_coef$estimate, col = "green", names.arg = rownames(mean_coef$term), main = "Mean
Imputation Estimates", ylab = "Estimate", las = 2)

barplot(multiple_coef$estimate, col = "red", names.arg = rownames(multiple_coef$term), main =
"Multiple Imputation Estimates", ylab = "Estimate", las = 2)
```



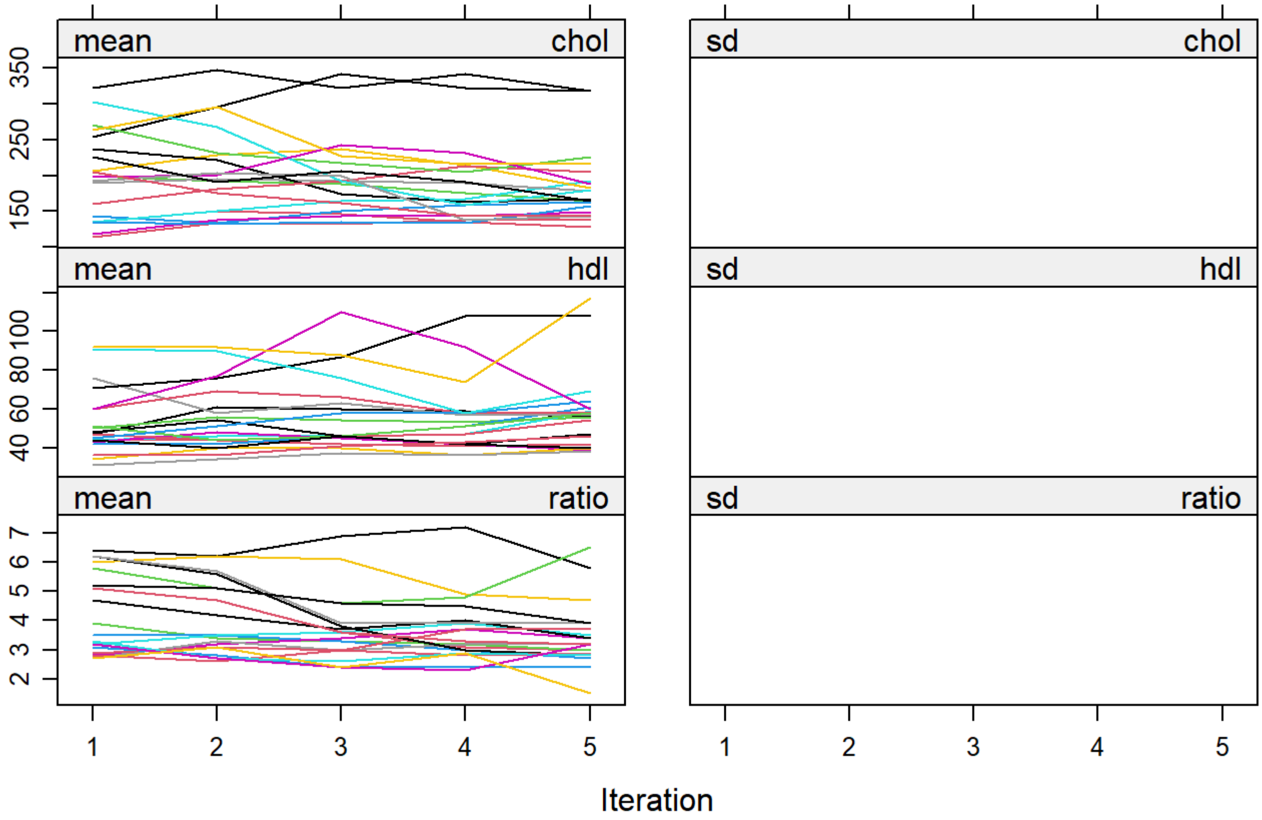
```
# Standard error comparison
barplot(complete_case_coef[,2], col = "blue", names.arg = rownames(complete_case_coef), main = "Complete Case SE")

barplot(mean_coef[,2], col = "green", names.arg = rownames(mean_coef), main = "Mean Imputation SE")

barplot(multiple_coef[,2], col = "red", names.arg = rownames(multiple_coef), main = "Multiple Imputation SE")
```



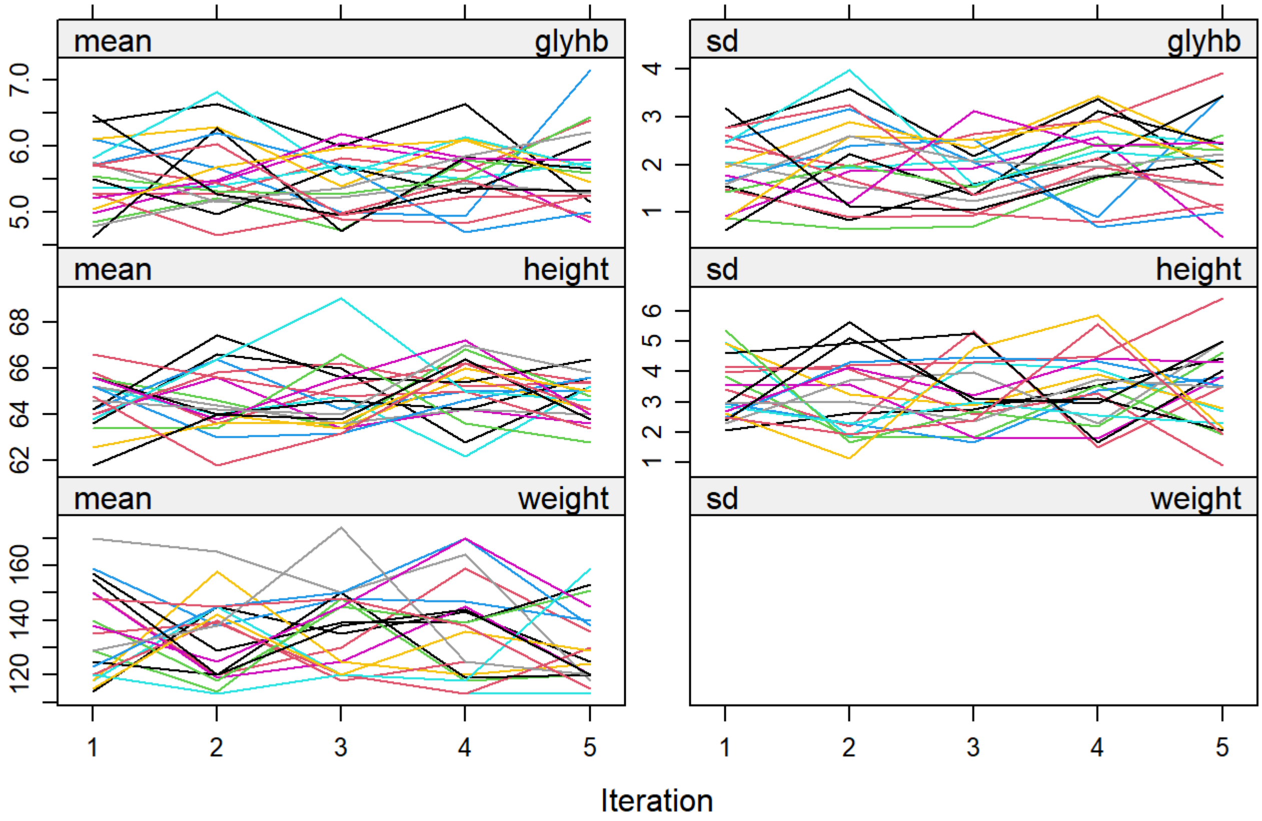
Step 8: Diagnostics

```
# Convergence plot
plot(C.imp, main = "Convergence Plot for MICE Imputation")
```

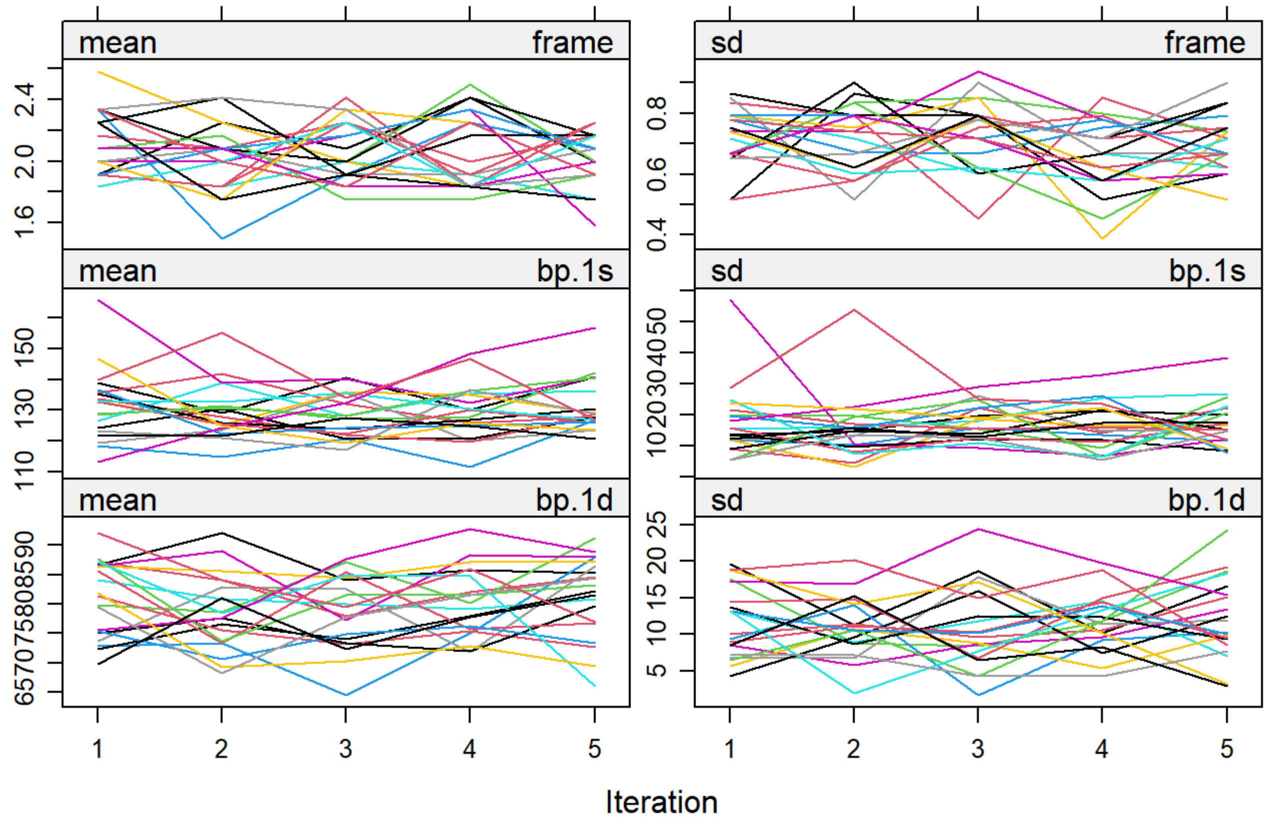
Convergence Plot for MICE Imputation



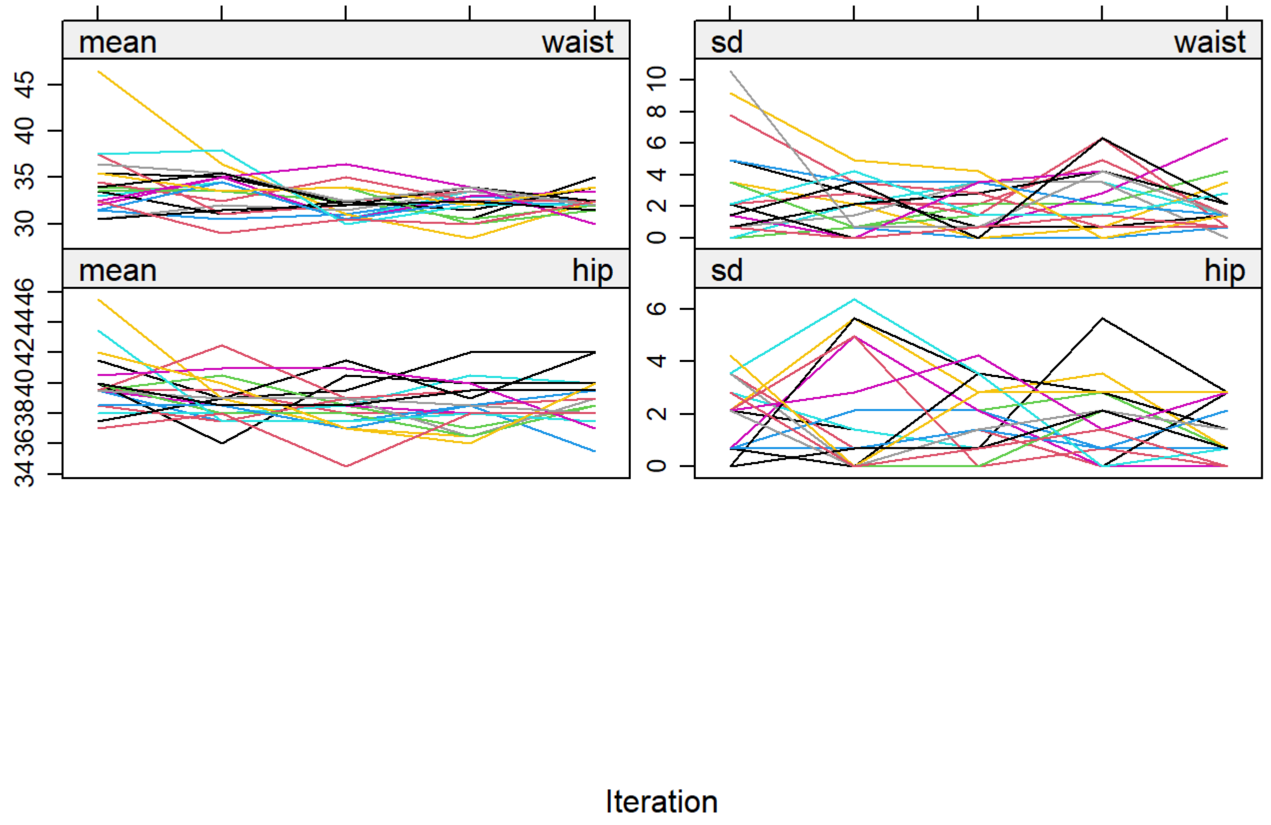
Convergence Plot for MICE Imputation



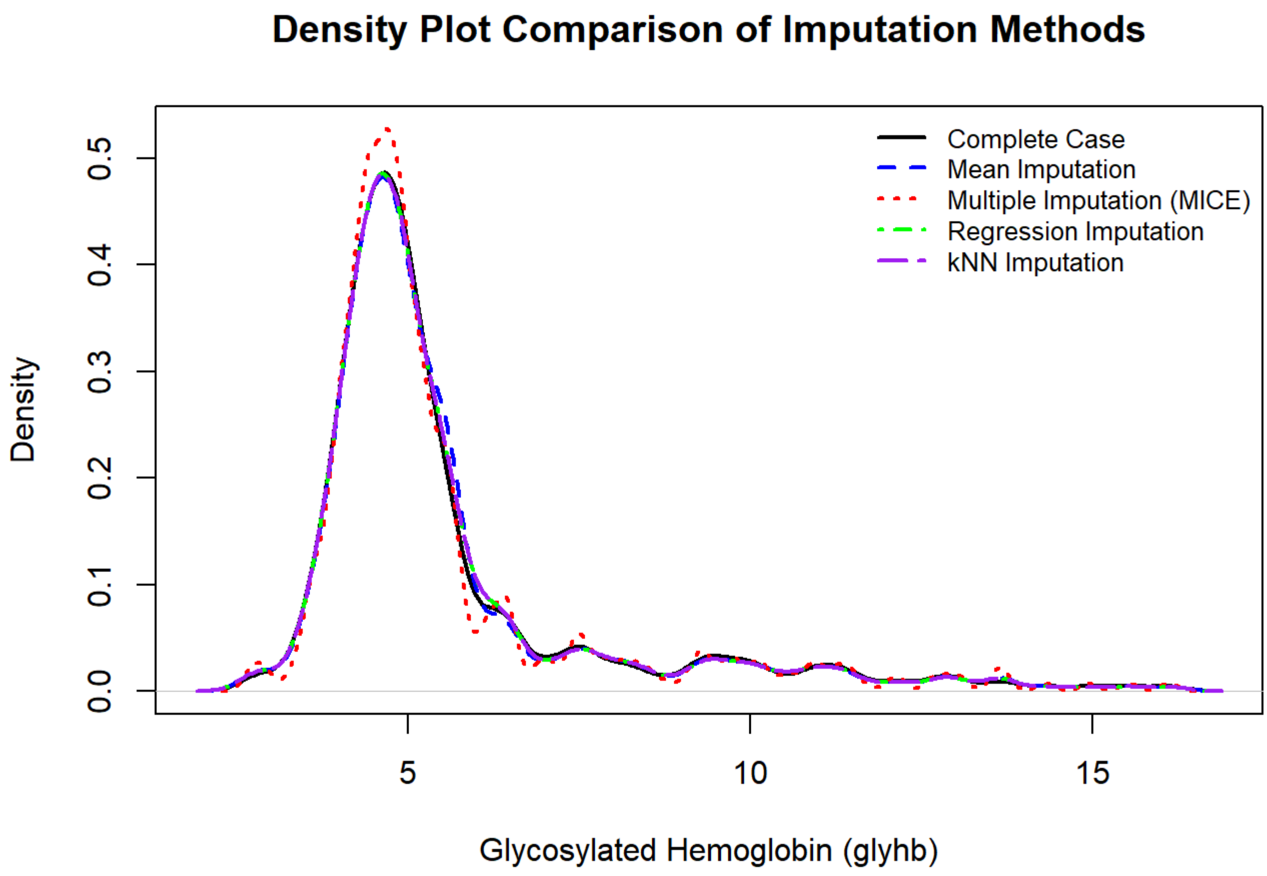
Convergence Plot for MICE Imputation



Convergence Plot for MICE Imputation



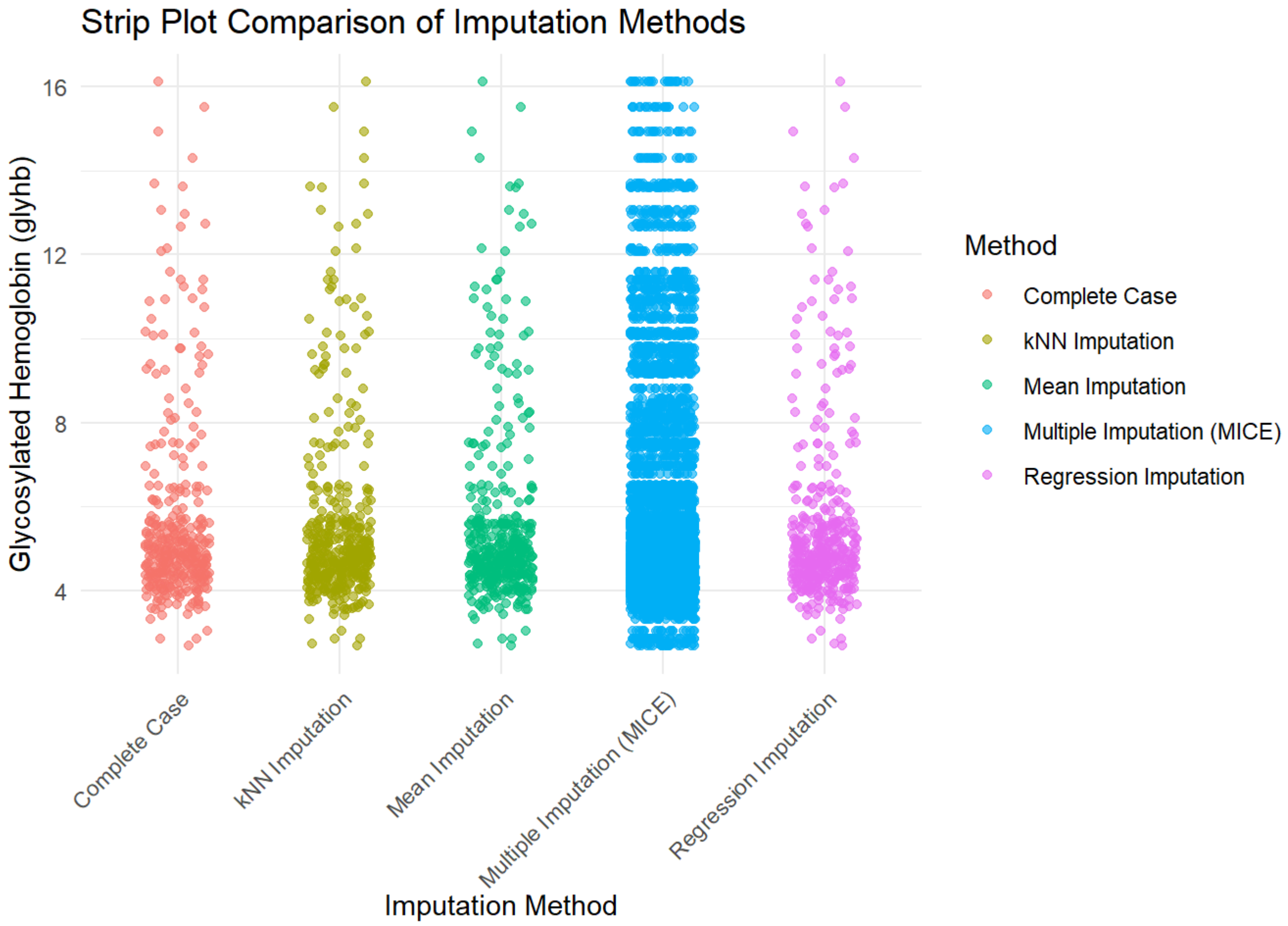
```
# Diagnostic plots

## Density plot
# Calculate density for the complete case analysis
complete_case_density <- density(DM1$glyhb, na.rm = TRUE)

# Calculate density for mean imputation
mean_imputed_data <- complete(M.imp, 1) # Extract the mean-imputed dataset
mean_imp_density <- density(mean_imputed_data$glyhb, na.rm = TRUE)

# Calculate density for multiple imputation using MICE
mice_imputed_data <- complete(C.imp, "long") # Extract the first complete imputed dataset
multiple_imp_density <- density(mice_imputed_data$glyhb, na.rm = TRUE)

# Calculate density for regression imputation
reg_imputed_data <- DM # This is the dataset where regression imputation has replaced missing glyhb
reg_imp_density <- density(reg_imputed_data$glyhb, na.rm = TRUE)

# Calculate density for kNN imputation
knn_imputed_data <- DM_knn # Dataset with kNN imputation applied
knn_imp_density <- density(knn_imputed_data$glyhb, na.rm = TRUE)

# Create the plot
plot(complete_case_density, col = "black", lwd = 2, lty = 1,
     main = "Density Plot Comparison of Imputation Methods",
     xlab = "Glycosylated Hemoglobin (glyhb)",
     ylab = "Density",
     ylim = c(0, max(c(complete_case_density$y, mean_imp_density$y,
                       multiple_imp_density$y, reg_imp_density$y,
                       knn_imp_density$y))))
lines(mean_imp_density, col = "blue", lwd = 2, lty = 2)      # Mean Imputation
lines(multiple_imp_density, col = "red", lwd = 2, lty = 3)   # Multiple Imputation (MICE)
lines(reg_imp_density, col = "green", lwd = 2, lty = 4)      # Regression Imputation
lines(knn_imp_density, col = "purple", lwd = 2, lty = 5)     # kNN Imputation

# Add a legend
legend("topright", legend = c("Complete Case", "Mean Imputation",
                              "Multiple Imputation (MICE)", "Regression Imputation",
                              "kNN Imputation"),
       col = c("black", "blue", "red", "green", "purple"),
       lty = c(1, 2, 3, 4, 5),
       lwd = 2, bty = "n", cex = 0.8)
```



```
## Strip plot
# Combine data into a single dataset for visualization
glyhb_combined <- data.frame(
  glyhb = c(DM1$glyhb,                         # Complete Case
            mean_imputed_data$glyhb,         # Mean Imputation
            mice_imputed_data$glyhb,         # Multiple Imputation (MICE)
            reg_imputed_data$glyhb,          # Regression Imputation
            knn_imputed_data$glyhb),         # kNN Imputation
  Method = factor(rep(c("Complete Case", "Mean Imputation",
                        "Multiple Imputation (MICE)", "Regression Imputation",
                        "kNN Imputation"),
                      times = c(length(DM1$glyhb),
                                length(mean_imputed_data$glyhb),
                                length(mice_imputed_data$glyhb),
                                length(reg_imputed_data$glyhb),
                                length(knn_imputed_data$glyhb))))
)

# Create the strip plot
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
ggplot(glyhb_combined, aes(x = Method, y = glyhb, color = Method)) +
  geom_jitter(width = 0.2, alpha = 0.6) +
  theme_minimal() +
  labs(
    title = "Strip Plot Comparison of Imputation Methods",
    x = "Imputation Method",
    y = "Glycosylated Hemoglobin (glyhb)"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

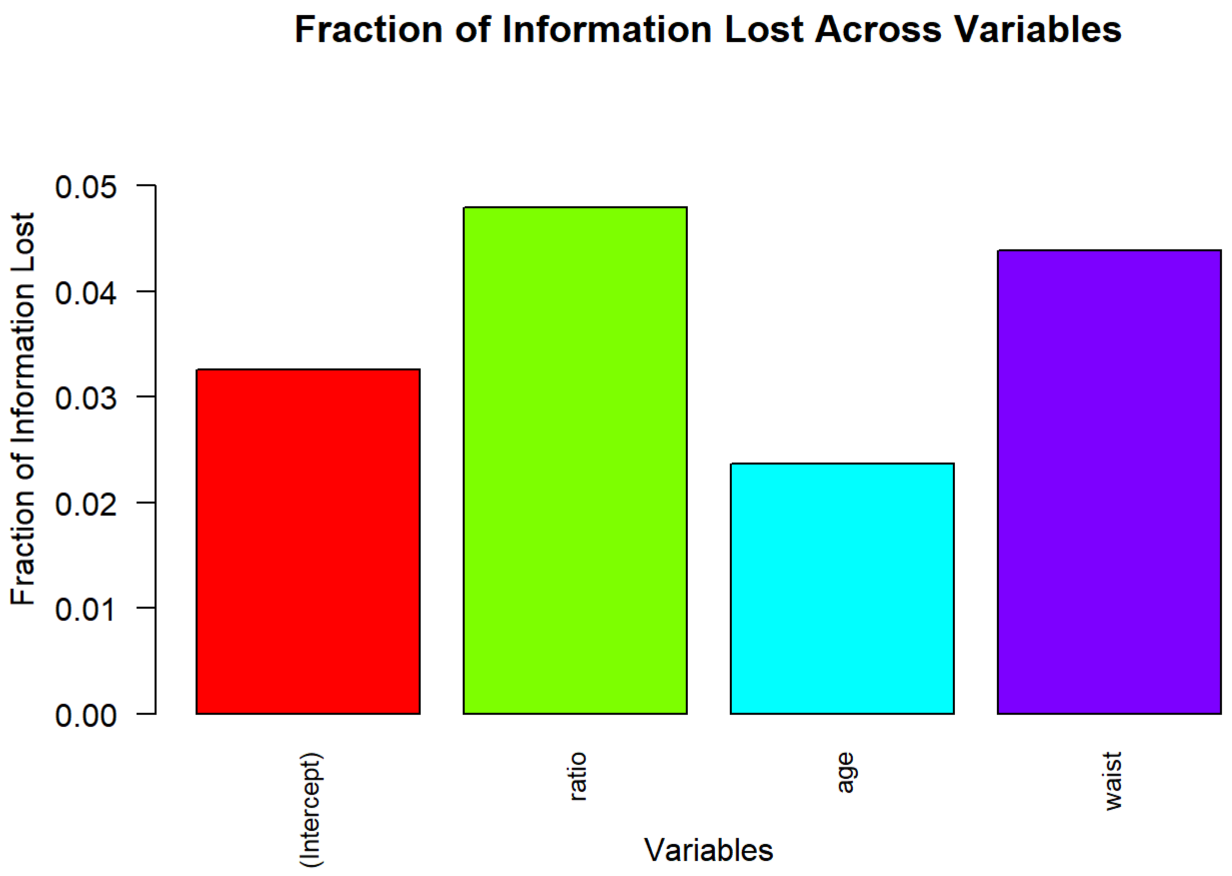


Step 9: Information Loss Analysis

```
# Extract coefficients from multiple imputation analyses
# Model8 is the final reduced model from multiple imputations
NI <- 20  # Number of imputations
beta_list <- lapply(model8$analyses, coefficients)  # Extract coefficients from each imputed
model

# Create a matrix for coefficients
beta_matrix <- do.call(rbind, beta_list)  # Combine into a matrix where each row is one imput
ation's coefficients

# Calculate between-imputation variability (B)
B <- cov(beta_matrix)

# Extract within-imputation variability (W)
Cov_list <- lapply(model8$analyses, vcov)  # Extract covariance matrices for each imputation
W <- Reduce("+", Cov_list) / NI  # Average covariance matrix (within-imputation variability)

# Calculate total variability (T)
T <- W + B  # Total variability

# Calculate fraction of information lost
info_loss <- diag(B) / diag(T)  # Fraction of information lost for each variable
round(info_loss, 3)  # Display rounded results
```

```
## (Intercept)       ratio         age       waist 
##       0.033       0.048       0.024       0.044
```

```
# Barplot for Information Loss
barplot(info_loss, col = rainbow(length(info_loss)),
        main = "Fraction of Information Lost Across Variables",
        xlab = "Variables",
        ylab = "Fraction of Information Lost",
        ylim = c(0, max(info_loss) * 1.2),
        names.arg = names(info_loss),
        las = 2, cex.names = 0.8)
```


Fraction of Information Lost Across Variables
Fraction of Information Lost
0.05
0.04
0.03
0.02
0.01
0.00
(Intercept)
ratio
age
waist
Variables