Task1 Seminar1

Group A3

2025-10-06

Task Introduction

Task 1: DiGeHormone - Investigating Factors Influencing Gastric Emptying in Type II Diabetes Mellitus A large cross-sectional study, named DiGeHormone, has been carried out in a population of individuals suffering from Type II Diabetes Mellitus (T2DM). The aim of the study is to investigate the association between a set of endogenous gastrointestinal (GI) hormones, T2DM disease factors and gastric emptying (GE).

The dataset (Data_T1.csv Download Data_T1.csv) includes basic demographic information, T2DM related factors, and GI biomarkers in 450 individuals with a T2DM diagnosis. A full description of the variables follow in Table 1 below.

Data Exploration

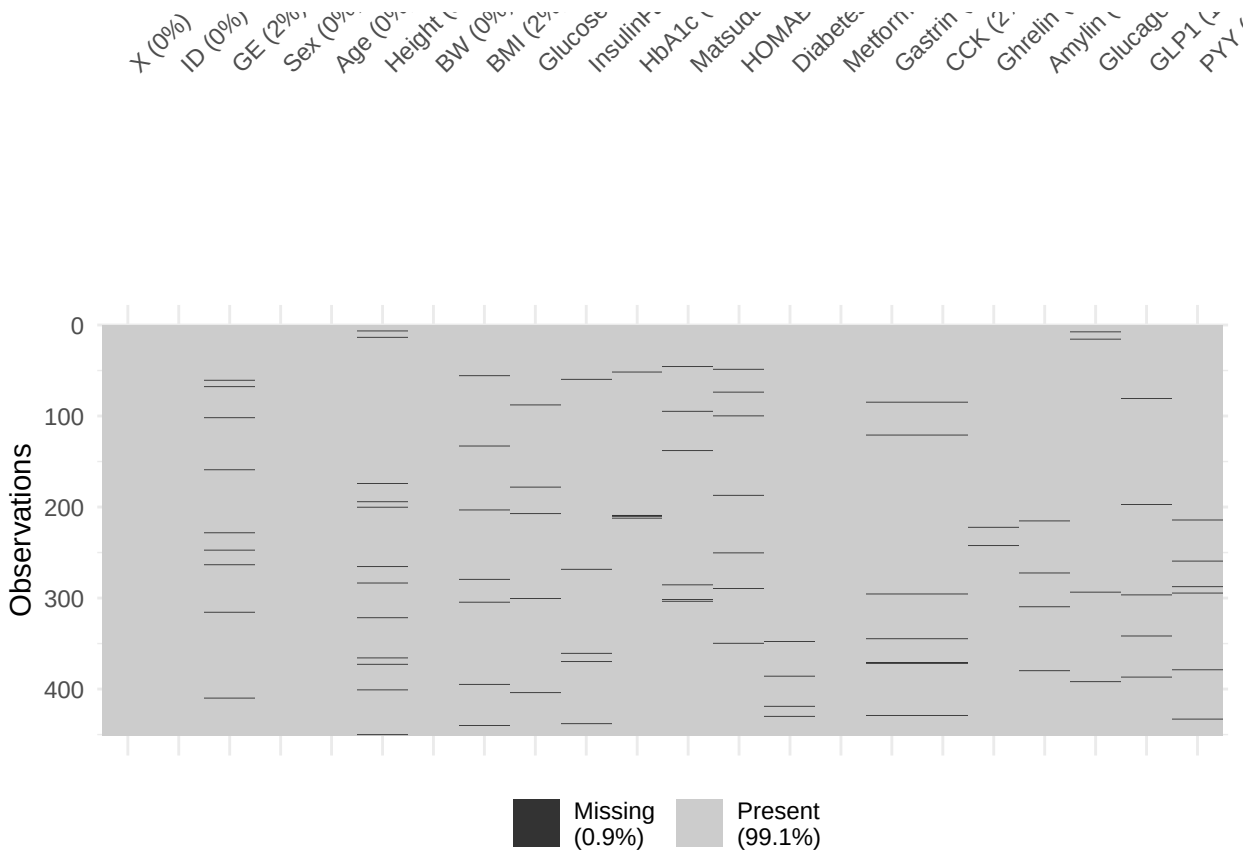
Step 1: data loading We load the data and visualized GE data.

```
### load data
dat <- read.csv("Data_T1.csv")
## showing data
head(dat)
```

```
##   X ID   GE Sex Age Height BW   BMI GlucoseFasting InsulinFasting HbA1c
## 1 1  1 29.8   1  40    164 90 33.46          9.620          9.645  8.08
## 2 2  2 53.2   1  35    159 59 23.34          9.833          6.760  8.26
## 3 3  3 44.4   0  48    174 81 26.75          8.717          9.307  8.74
## 4 4  4 58.8   0  37    154 65 27.41         11.179          6.948  8.89
## 5 5  5 54.9   0  47    179 65 20.29          8.613          5.198  7.89
## 6 6  6 37.6   0  55    179 82 25.59          9.461         10.941  8.47
##   MatsudaIdx   HOMAB DiabetesComplications Metformin Gastrin   CCK Ghrelin
## 1       4.12 12.8500                     0         0   93.87 93.87   319.8
## 2       5.28  8.9750                     0         1   86.37 86.37   189.3
## 3       4.17 11.4050                     0         0   73.19 73.19   227.1
## 4       5.41  9.8900                     0         0   81.32 81.32   326.1
## 5       6.47  7.4175                     1         0   59.48 59.48   388.3
## 6       3.62 12.3500                     0         0   60.82 60.82   327.5
##   Amylin Glucagon GLP1   PYY
## 1  16.76    9.002 2.35 99.75
## 2  13.23   13.711 5.41 84.85
## 3  15.79   16.701 3.97 86.47
## 4  16.17    9.890 4.69 57.50
```

```
## 5  11.98   11.074 2.81 55.99
## 6  12.32   12.669 3.62 83.64
```

```
library(naniar)
vis_miss(dat)
```



As shown in the figure, approximately 1% of the data is missing in the dataset. Missing data is relatively few, so we can delete the missing lines in data analysis.

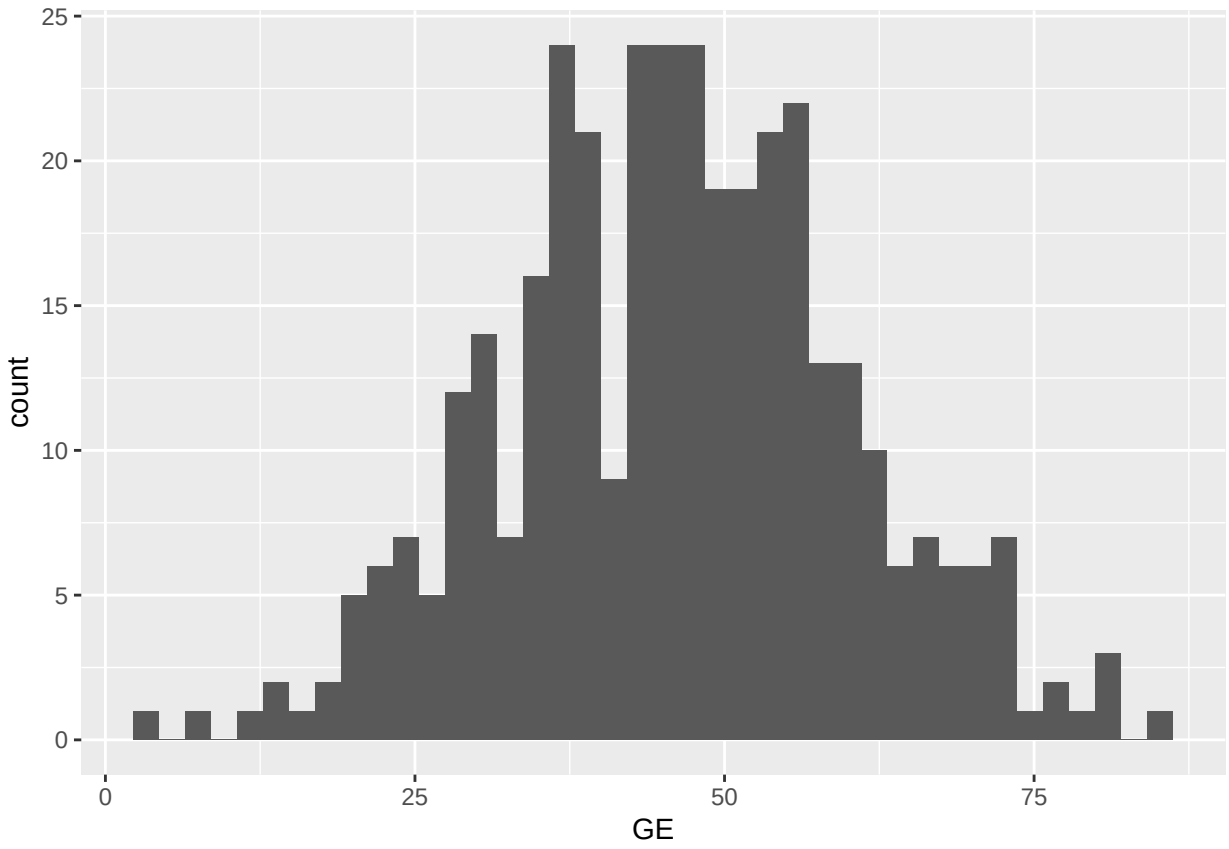
Step 2: Data exploring

We use visualizations and statistic tests to have an understanding of the data, in order to build suitable models. GE is shown below.

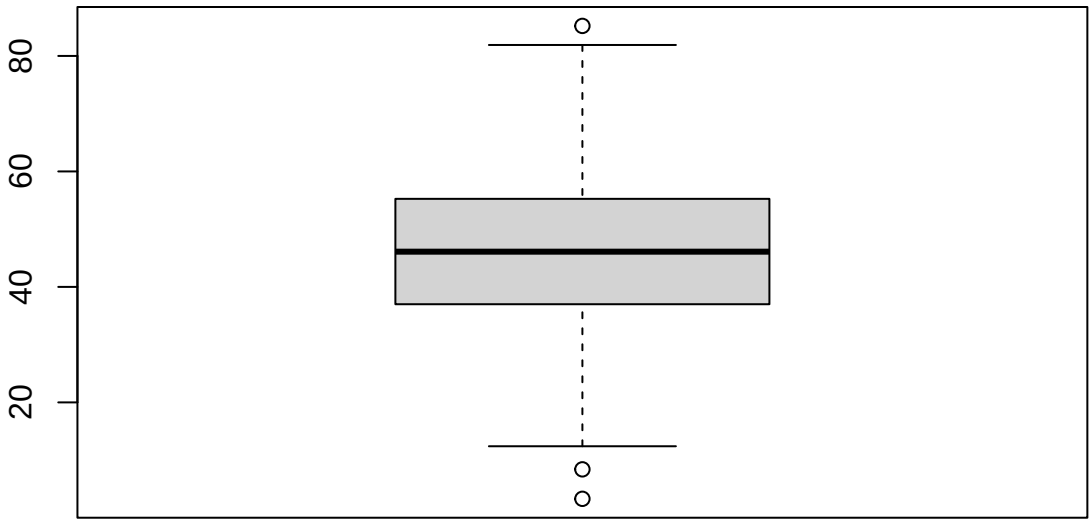
```
dat <- dat[complete.cases(dat), ]

library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)

ggplot(dat, aes(GE)) + geom_histogram(bins=40)
```




```
boxplot(dat$GE)
```



We devided the variables into 3 types as follow:

Biomarkers: Gastrin, CCK, Ghrelin, Amylin, Glucagon, GLP1, PYY. All values are continuous.

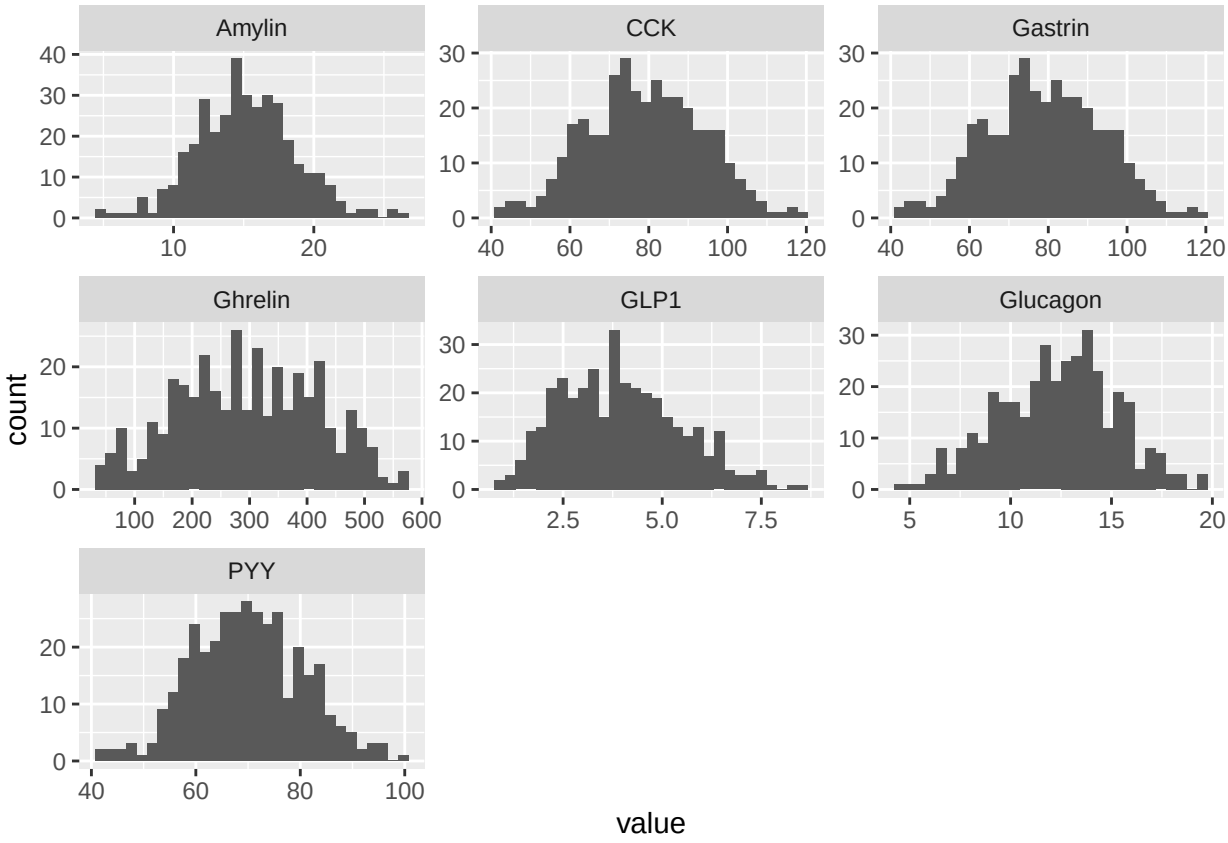
Continuous Variables: Age, BW, BMI, GlucoseFasting, InsulinFasting, HbA1c, MatsudaIdx, HOMAB. All values are continuous.

Categorical Variables: Sex, Metformin usage, DiabetesComplications. All values are 0-1.

First, we visualize variables with continuous values, which is Biomarkers and Continuous Variables.

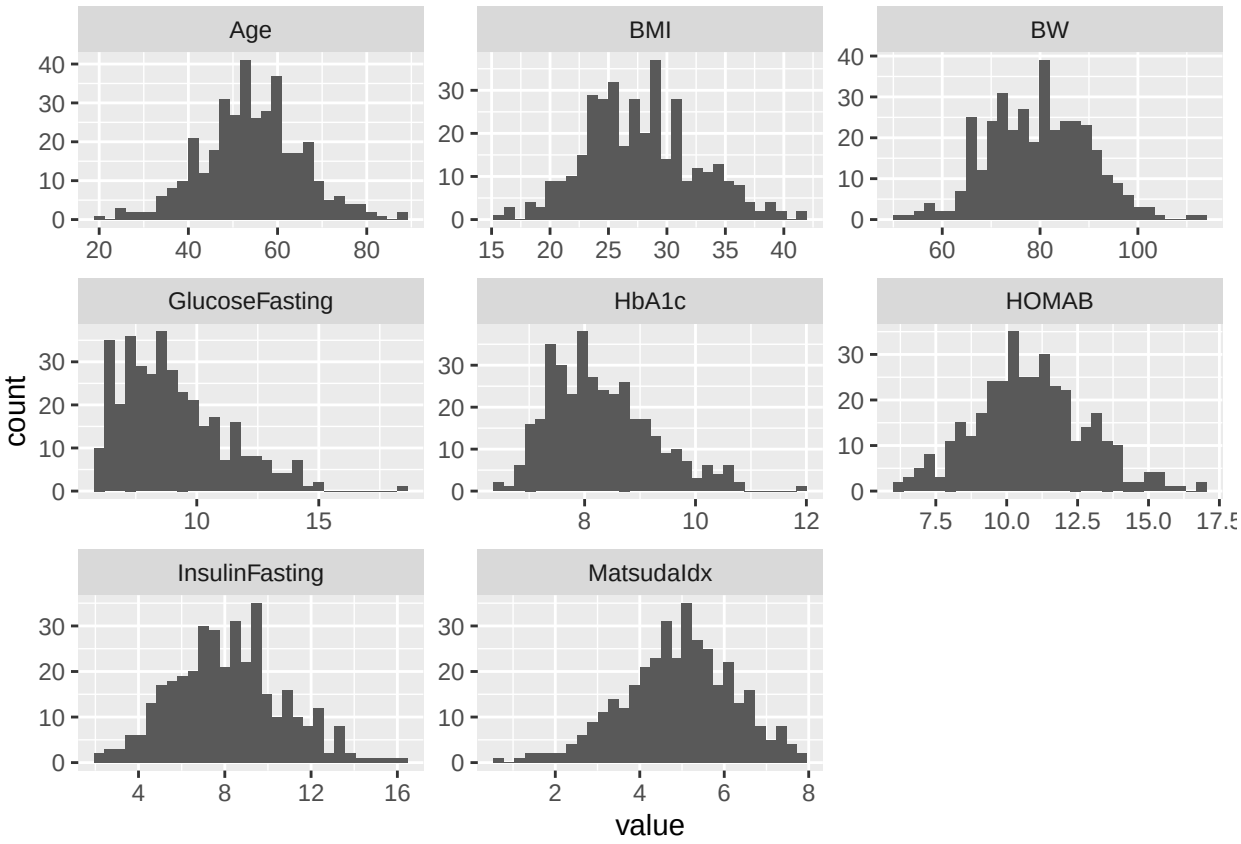
```
# histogram of biomarkers & contimuous_vars
library(tidyr)
biomarkers <- c("Gastrin","CCK","Ghrelin","Amylin","Glucagon","GLP1","PYY")
continuous_vars <- c("Age","BW","BMI","GlucoseFasting","InsulinFasting","HbA1c","MatsudaIdx","HOMAB")
categorical_vars <- c("Sex","Metformin","DiabetesComplications")

dat %>% select(all_of(biomarkers)) %>%
  pivot_longer(everything(), names_to="marker", values_to="value") %>%
  ggplot(aes(x=value)) + facet_wrap(~marker, scales="free") + geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value `binwidth`.
```




```
dat %>% select(all_of(continuous_vars)) %>%
  pivot_longer(everything(), names_to="marker", values_to="value") %>%
  ggplot(aes(x=value)) + facet_wrap(~marker, scales="free") + geom_histogram()
```

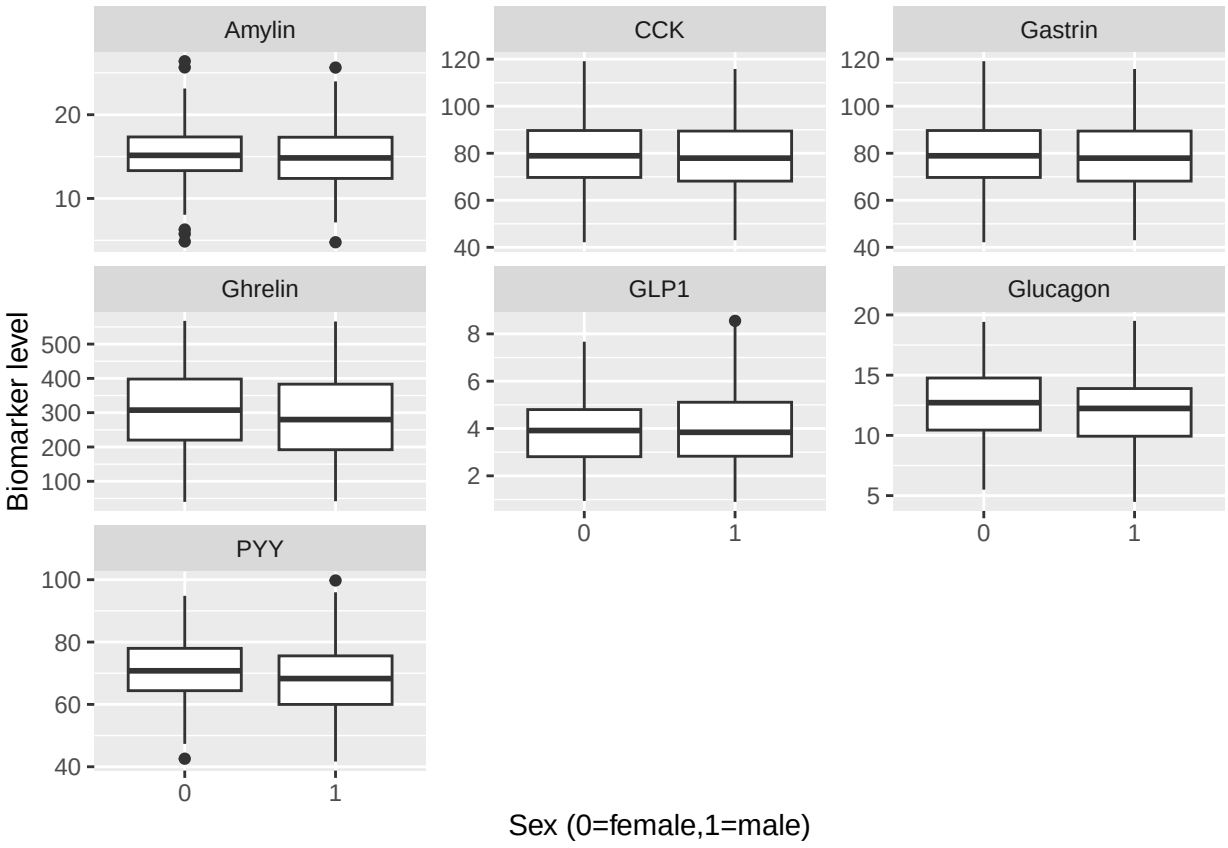
```
## `stat_bin()` using `bins = 30`. Pick better value `binwidth`.
```



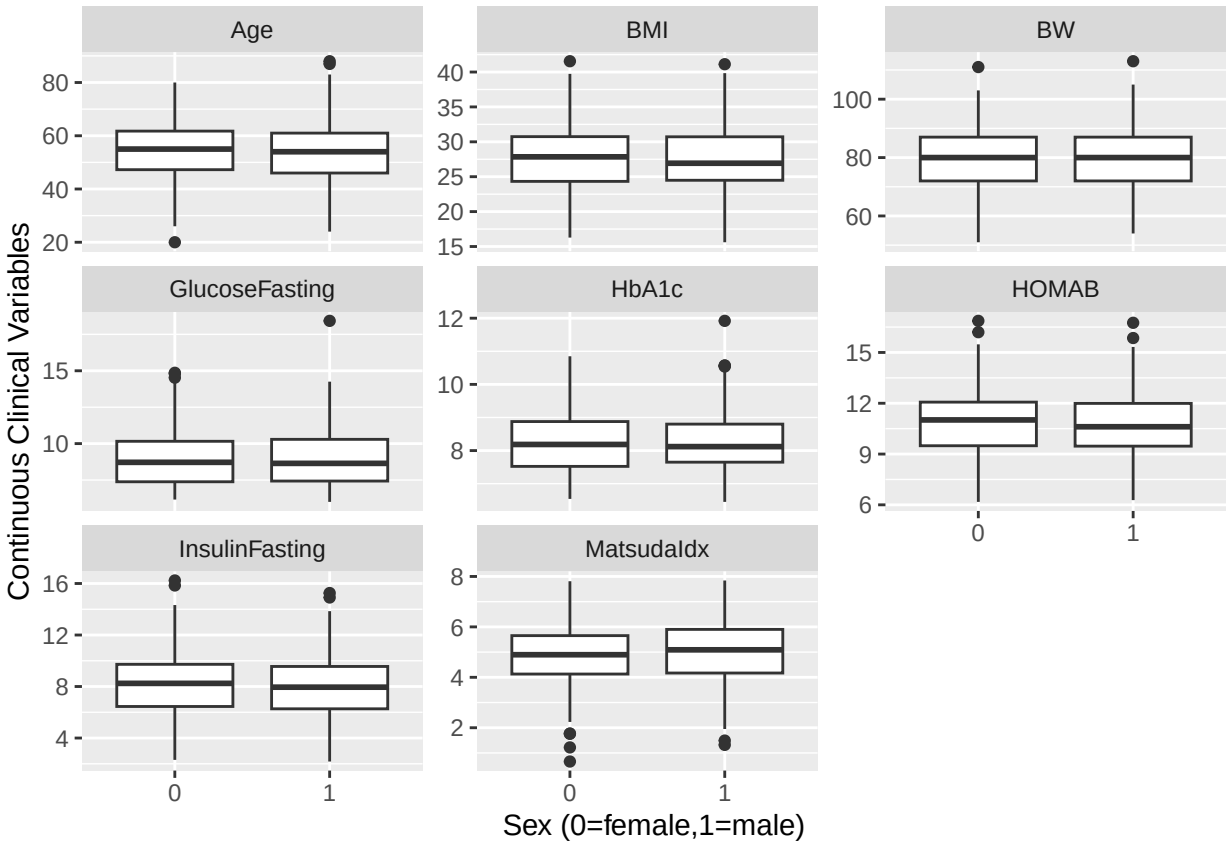
As shown if the figure, most biomarkers looks approximately normally distributed while some of the continuous variables looks scewed.

After that, we want to see if these biomarkers and other continuous variables are related to categorical variables of sex, metformin and diabetes complications. We did boxplots of variables with continuous values in different sex, metformin and diabetes complications, and ran statistic tests according to normality.

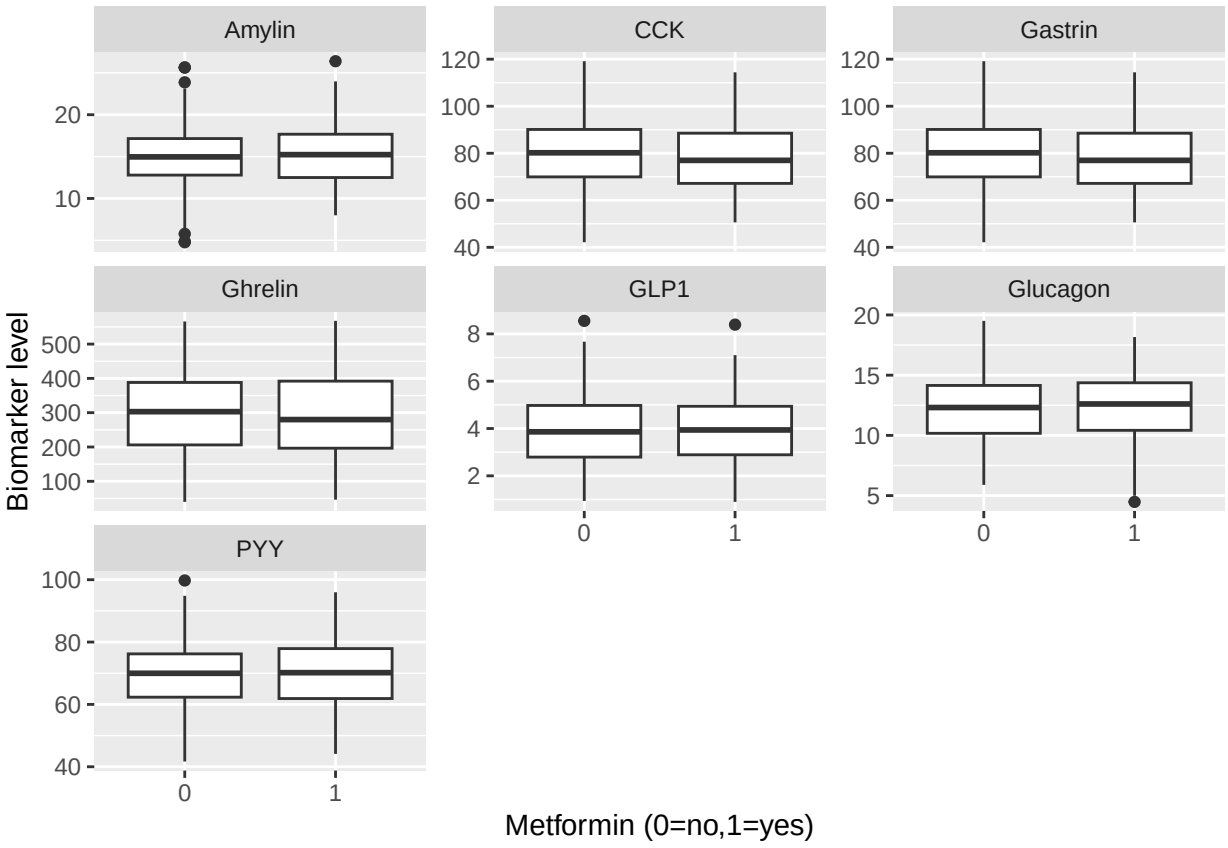
```
# biomarkers, continuous_vars grouped by sex, Metformin, DBComplications
dat %>%
  pivot_longer(cols = all_of(biomarkers), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(Sex), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "Sex (0=female,1=male)", y = "Biomarker level")
```



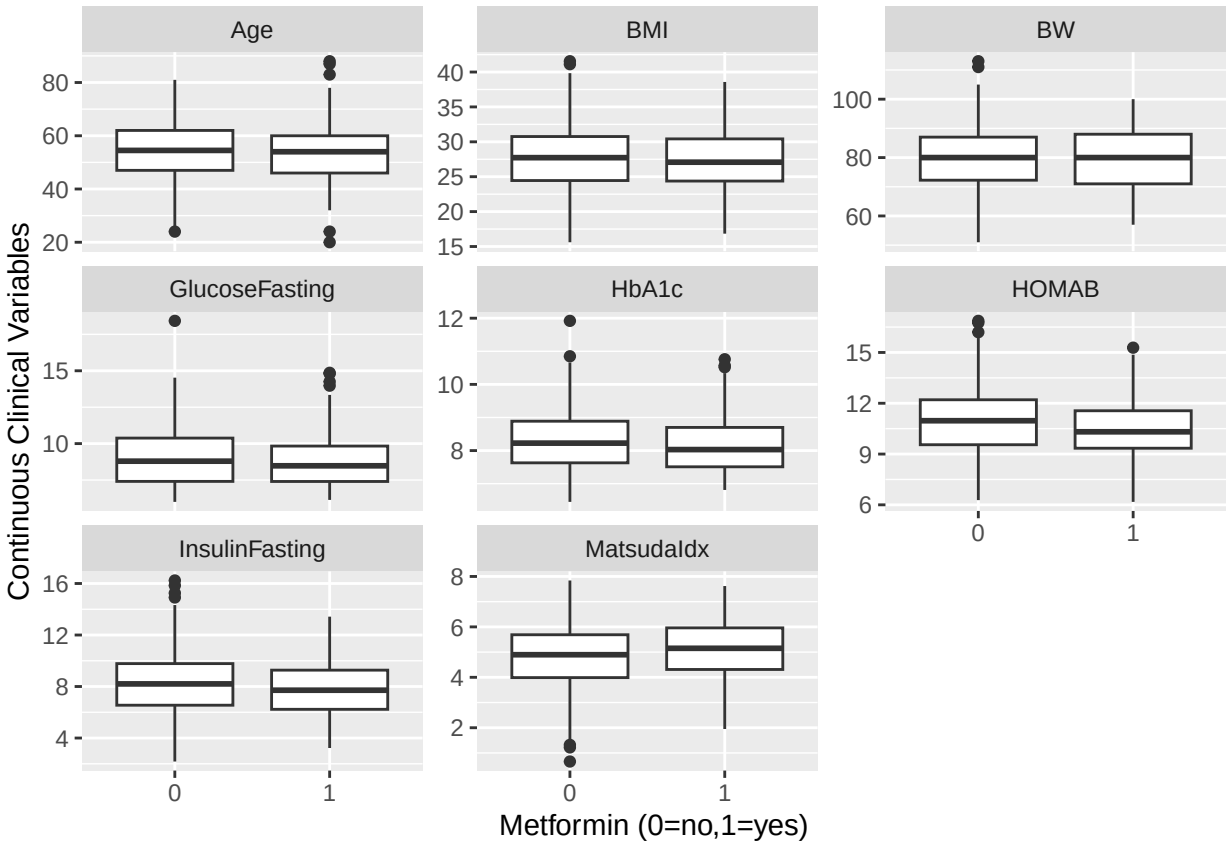

```
dat %>%
  pivot_longer(cols = all_of(continuous_vars), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(Sex), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "Sex (0=female,1=male)", y = "Continuous Clinical Variables")
```



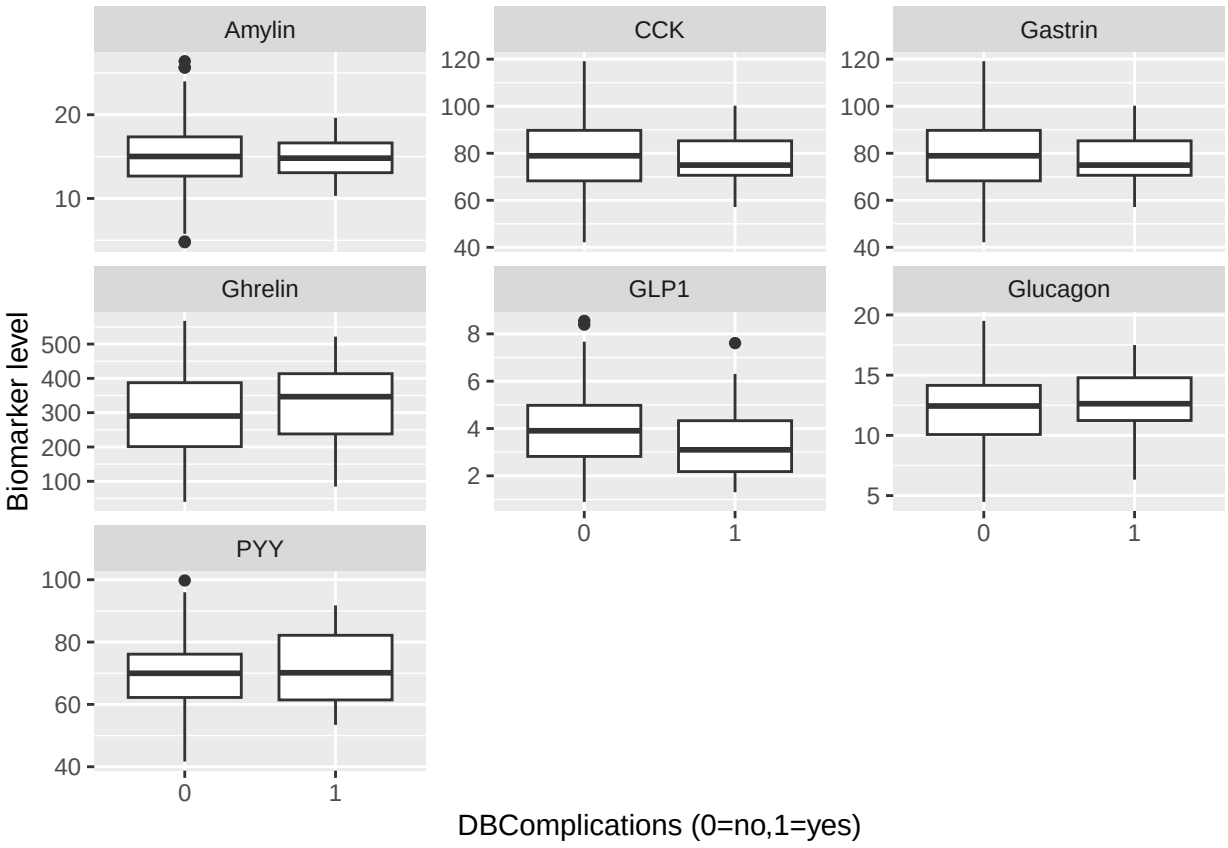

```
dat %>%
  pivot_longer(cols = all_of(biomarkers), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(Metformin), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "Metformin (0=no,1=yes)", y = "Biomarker level")
```



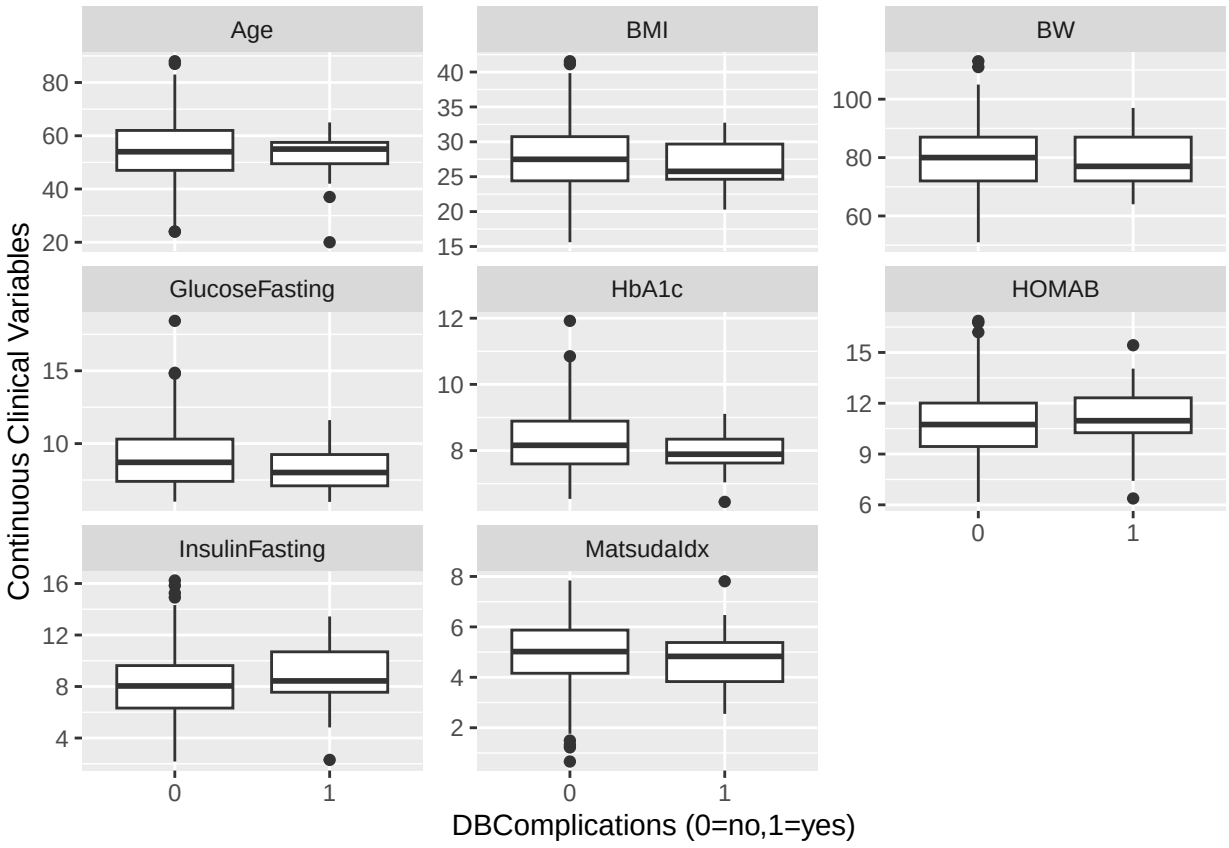

```
dat %>%
  pivot_longer(cols = all_of(continuous_vars), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(Metformin), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "Metformin (0=no,1=yes)", y = "Continuous Clinical Variables")
```




```
dat %>%
  pivot_longer(cols = all_of(biomarkers), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(DiabetesComplications), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "DBComplications (0=no,1=yes)", y = "Biomarker level")
```




```
dat %>%
  pivot_longer(cols = all_of(continuous_vars), names_to = "marker", values_to = "value") %>%
  ggplot(aes(x = factor(DiabetesComplications), y = value)) +
  geom_boxplot() +
  facet_wrap(~marker, scales = "free_y") +
  labs(x = "DBComplications (0=no,1=yes)", y = "Continuous Clinical Variables")
```




```r
# normalization check
all_vars <- c(biomarkers, continuous_vars)
results <- data.frame(Group = character(),
                      Variable = character(),
                      Test = character(),
                      P_value = numeric())

for (g in categorical_vars) {
  for (v in all_vars) {
    x <- dat[[v]]
    group <- dat[[g]]
    df <- data.frame(x, group)
    df <- df[complete.cases(df), ]
    x <- df$x
    group <- df$group
    sw1 <- shapiro.test(x[group == unique(group)[1]])$p.value
    sw2 <- shapiro.test(x[group == unique(group)[2]])$p.value
    if (sw1 > 0.05 & sw2 > 0.05) {
      p <- t.test(x ~ group)$p.value
      test_used <- "t-test"
    } else {
      p <- wilcox.test(x ~ group)$p.value
      test_used <- "Wilcoxon"
    }
    results <- rbind(results, data.frame(Group = g,
                                         Variable = v,
                                         Test = test_used,
```

```
                                        P_value = round(p, 4)))
  }
}
print(results)
```

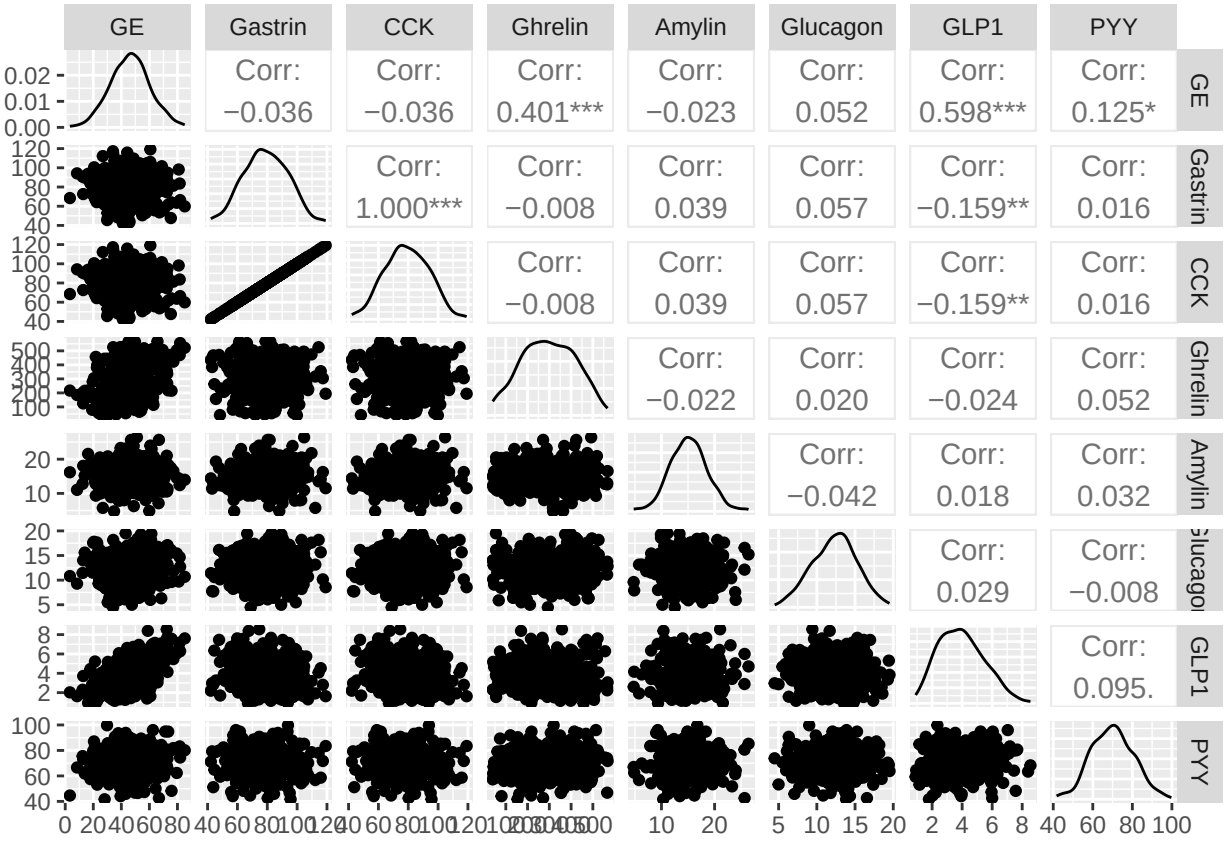
```
##                    Group       Variable     Test P_value
## 1                    Sex        Gastrin   t-test  0.5444
## 2                    Sex            CCK   t-test  0.5444
## 3                    Sex        Ghrelin Wilcoxon  0.1134
## 4                    Sex         Amylin   t-test  0.3241
## 5                    Sex       Glucagon   t-test  0.0456
## 6                    Sex           GLP1 Wilcoxon  0.8351
## 7                    Sex            PYY   t-test  0.0268
## 8                    Sex            Age   t-test  0.5273
## 9                    Sex             BW   t-test  0.7198
## 10                   Sex            BMI   t-test  0.6742
## 11                   Sex GlucoseFasting Wilcoxon  0.8281
## 12                   Sex InsulinFasting   t-test  0.4974
## 13                   Sex          HbA1c Wilcoxon  0.9721
## 14                   Sex     MatsudaIdx   t-test  0.3282
## 15                   Sex          HOMAB   t-test  0.4467
## 16             Metformin        Gastrin   t-test  0.4116
## 17             Metformin            CCK   t-test  0.4116
## 18             Metformin        Ghrelin Wilcoxon  0.4346
## 19             Metformin         Amylin   t-test  0.5009
## 20             Metformin       Glucagon Wilcoxon  0.7983
## 21             Metformin           GLP1 Wilcoxon  0.6293
## 22             Metformin            PYY   t-test  0.8813
## 23             Metformin            Age   t-test  0.7596
## 24             Metformin             BW   t-test  0.5189
## 25             Metformin            BMI   t-test  0.2562
## 26             Metformin GlucoseFasting Wilcoxon  0.2591
## 27             Metformin InsulinFasting   t-test  0.1080
## 28             Metformin          HbA1c Wilcoxon  0.3589
## 29             Metformin     MatsudaIdx   t-test  0.0920
## 30             Metformin          HOMAB   t-test  0.0640
## 31 DiabetesComplications        Gastrin   t-test  0.7495
## 32 DiabetesComplications            CCK   t-test  0.7495
## 33 DiabetesComplications        Ghrelin Wilcoxon  0.1605
## 34 DiabetesComplications         Amylin   t-test  0.6412
## 35 DiabetesComplications       Glucagon   t-test  0.5375
## 36 DiabetesComplications           GLP1 Wilcoxon  0.0761
## 37 DiabetesComplications            PYY   t-test  0.4216
## 38 DiabetesComplications            Age Wilcoxon  0.4659
## 39 DiabetesComplications             BW   t-test  0.8189
## 40 DiabetesComplications            BMI   t-test  0.1937
## 41 DiabetesComplications GlucoseFasting Wilcoxon  0.0986
## 42 DiabetesComplications InsulinFasting Wilcoxon  0.1969
## 43 DiabetesComplications          HbA1c Wilcoxon  0.1807
## 44 DiabetesComplications     MatsudaIdx   t-test  0.4673
## 45 DiabetesComplications          HOMAB   t-test  0.5828
```

As we can see in the result, Sex has significant impact on Glucagon and PYY, because p-value of statistic tests

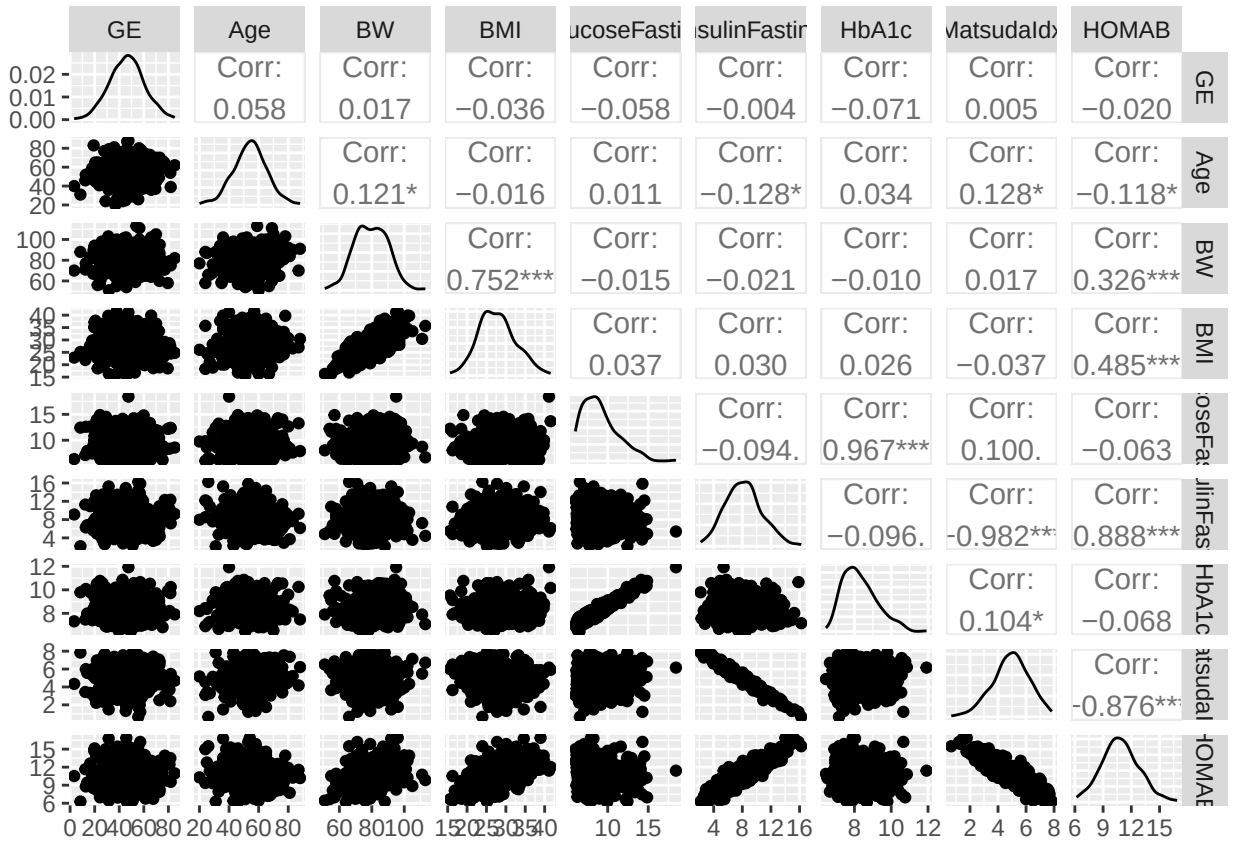
are lower than 0.05. Moreover, p values are between 0.05-1 of Metformin-MatsudaIdx, Metformin-HOMAB, DiabetesComplications-GLP1, and DiabetesComplications-GlucoseFasting.

Next, we examine whether variables with continuous value are correlated with each other.

```
# pair plot biomarkers & continuous_vars & GE
library(GGally)
selected <- dat %>% select(GE, all_of(biomarkers))
ggpairs(selected, columns=1:ncol(selected))
```



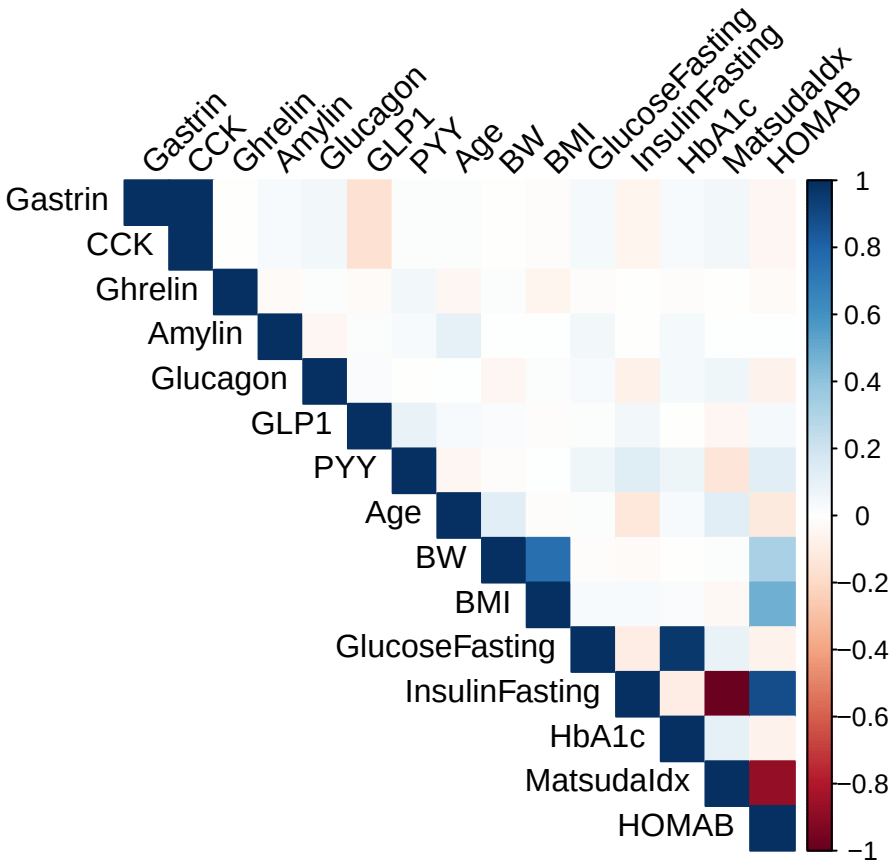

```
selected <- dat %>% select(GE, all_of(continuous_vars))
ggpairs(selected, columns=1:ncol(selected))
```




```
# heatbplot
library(corrplot)
```

```
## corrplot 0.95 loaded
```

```
mat <- cor(dat %>% select(all_of(biomarkers), all_of(continuous_vars)), use="pairwise.complete.obs")
corrplot(mat, method="color", type="upper", tl.col="black", tl.srt=45)
```




```
round(mat, 2)
```

```
##                Gastrin   CCK Ghrelin Amylin Glucagon  GLP1   PYY   Age    BW
## Gastrin           1.00  1.00   -0.01   0.04     0.06 -0.16  0.02  0.02 -0.01
## CCK               1.00  1.00   -0.01   0.04     0.06 -0.16  0.02  0.02 -0.01
## Ghrelin          -0.01 -0.01    1.00  -0.02     0.02 -0.02  0.05 -0.04  0.02
## Amylin            0.04  0.04   -0.02   1.00    -0.04  0.02  0.03  0.11  0.01
## Glucagon          0.06  0.06    0.02  -0.04     1.00  0.03 -0.01  0.00 -0.04
## GLP1             -0.16 -0.16   -0.02   0.02     0.03  1.00  0.09  0.03  0.02
## PYY               0.02  0.02    0.05   0.03    -0.01  0.09  1.00 -0.05 -0.01
## Age               0.02  0.02   -0.04   0.11     0.00  0.03 -0.05  1.00  0.12
## BW               -0.01 -0.01    0.02   0.01    -0.04  0.02 -0.01  0.12  1.00
## BMI              -0.01 -0.01   -0.05   0.00     0.01 -0.01  0.01 -0.02  0.75
## GlucoseFasting    0.04  0.04   -0.01   0.05     0.04  0.02  0.06  0.01 -0.01
## InsulinFasting   -0.05 -0.05    0.00   0.00    -0.08  0.06  0.13 -0.13 -0.02
## HbA1c             0.03  0.03   -0.02   0.04     0.04 -0.01  0.08  0.03 -0.01
## MatsudaIdx        0.05  0.05   -0.01   0.01     0.06 -0.05 -0.14  0.13  0.02
## HOMAB            -0.05 -0.05   -0.02   0.00    -0.06  0.04  0.12 -0.12  0.33
##                  BMI GlucoseFasting InsulinFasting HbA1c MatsudaIdx HOMAB
## Gastrin        -0.01           0.04          -0.05  0.03       0.05 -0.05
## CCK            -0.01           0.04          -0.05  0.03       0.05 -0.05
## Ghrelin        -0.05          -0.01           0.00 -0.02      -0.01 -0.02
## Amylin          0.00           0.05           0.00  0.04       0.01  0.00
## Glucagon        0.01           0.04          -0.08  0.04       0.06 -0.06
## GLP1           -0.01           0.02           0.06 -0.01      -0.05  0.04
## PYY             0.01           0.06           0.13  0.08      -0.14  0.12
```

```
## Age              -0.02          0.01          -0.13  0.03        0.13 -0.12
## BW                0.75         -0.01          -0.02 -0.01        0.02  0.33
## BMI               1.00          0.04           0.03  0.03       -0.04  0.48
## GlucoseFasting    0.04          1.00          -0.09  0.97        0.10 -0.06
## InsulinFasting    0.03         -0.09           1.00 -0.10       -0.98  0.89
## HbA1c             0.03          0.97          -0.10  1.00        0.10 -0.07
## MatsudaIdx       -0.04          0.10          -0.98  0.10        1.00 -0.88
## HOMAB             0.48         -0.06           0.89 -0.07       -0.88  1.00
```

From the figures, we can get the following conclusion:

There is a positive correlation between Body Weight-BMI, Glucose Faster-HbA1c, Insulin Fasting-HOMAB.

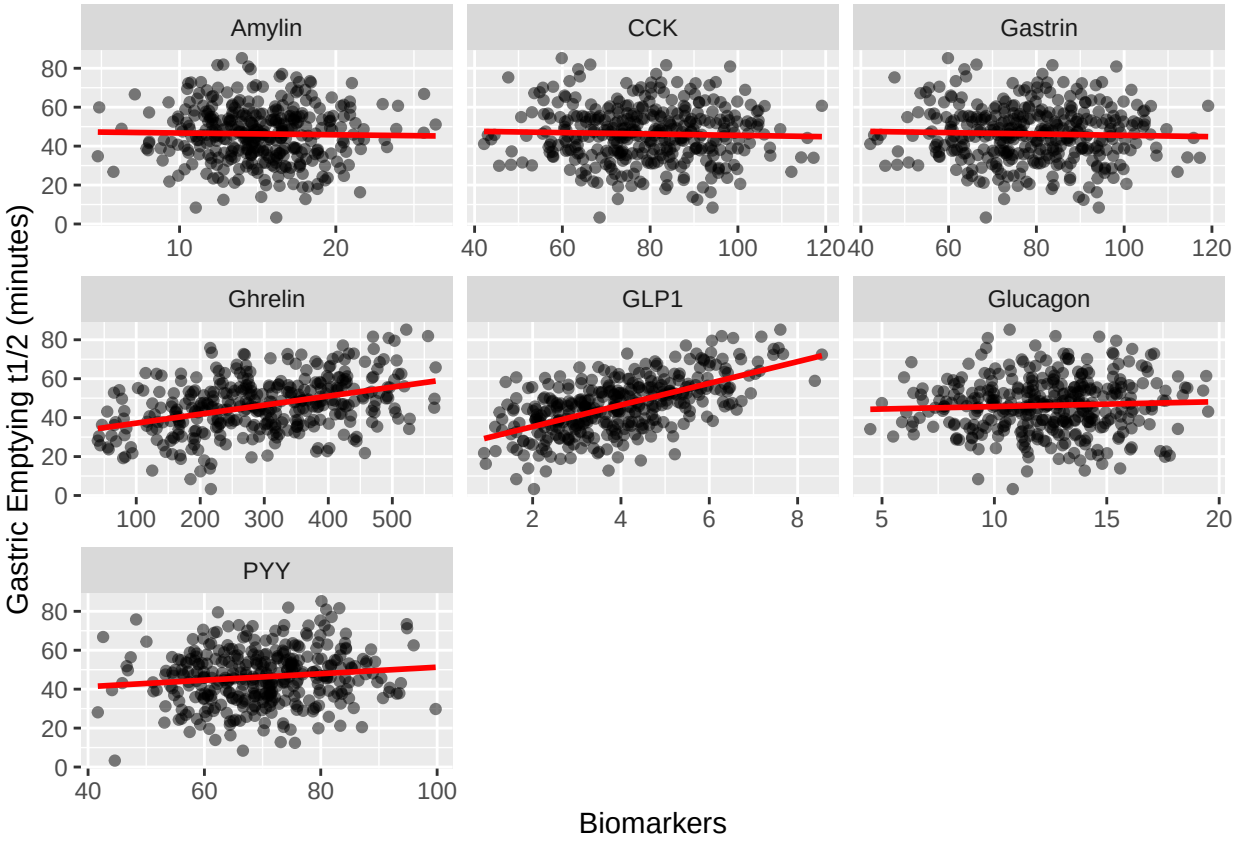
There is a negetive corraltion between Insulin Fasting-MatsudaIdx, MatsudaIdx-HOMAB.

It seems like CCK shares the exact same value with Gastrin.

We also looked at variables separately to see their correlation with GE.

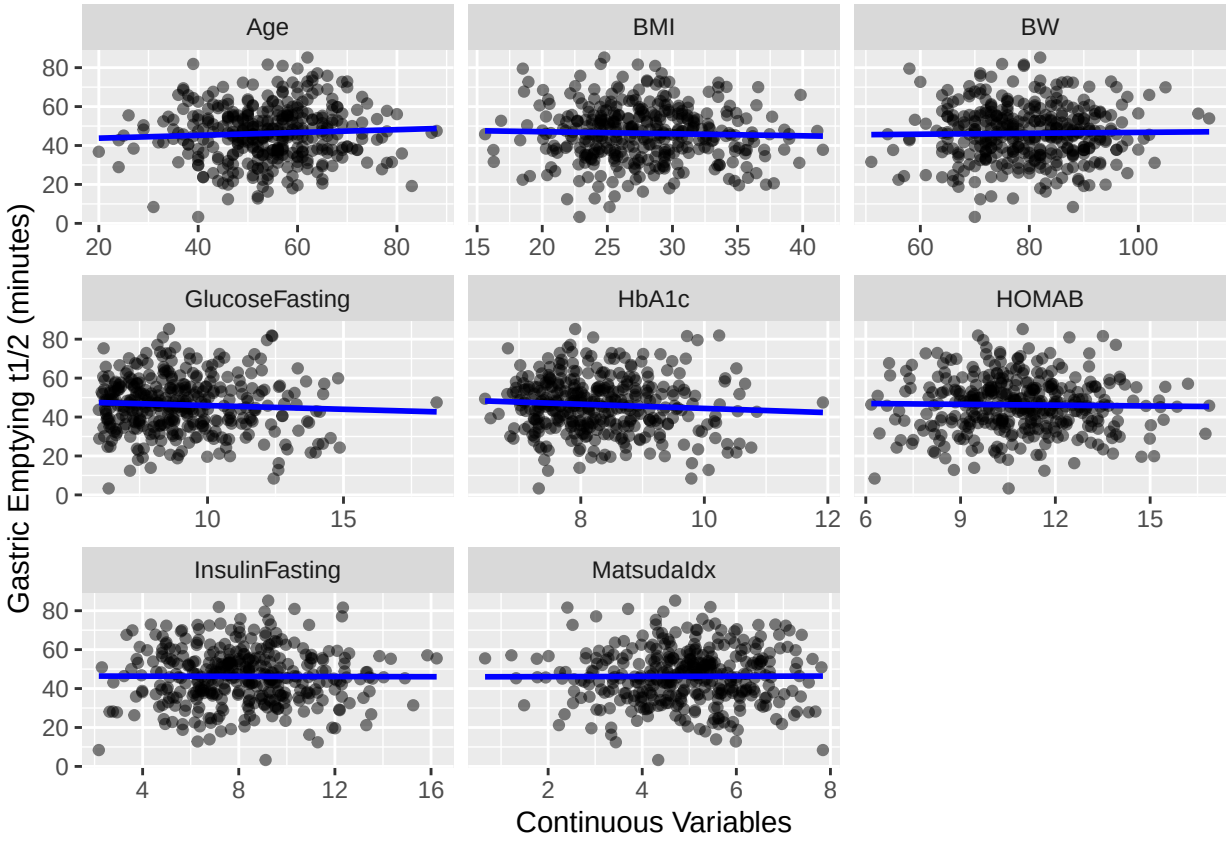
```
# scatter GE-biomarkers/continuous_vars
dat %>%
  pivot_longer(cols = all_of(biomarkers), names_to = "variable", values_to = "value") %>%
  ggplot(aes(x = value, y = GE)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  facet_wrap(~variable, scales = "free_x") +
  labs(x = "Biomarkers", y = "Gastric Emptying t1/2 (minutes)")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

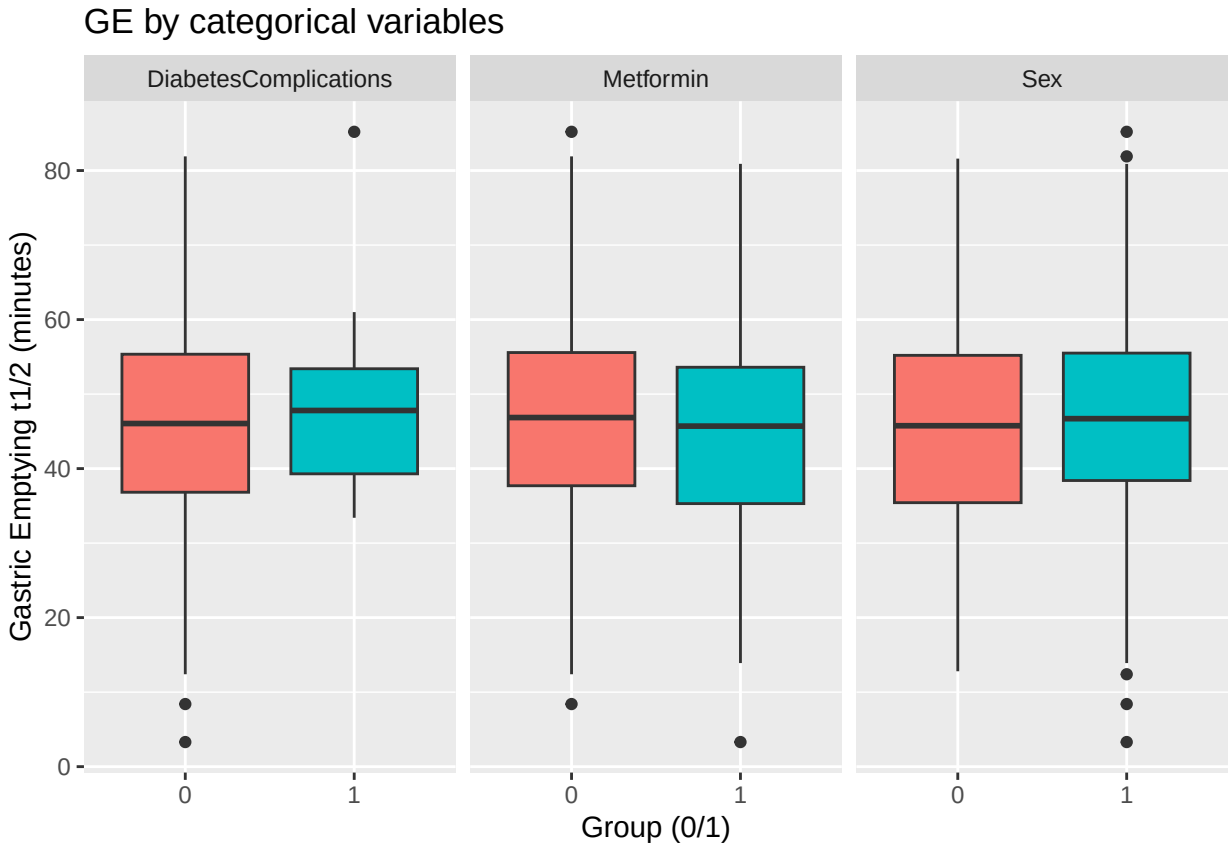


```
dat %>%
  pivot_longer(cols = all_of(continuous_vars), names_to = "variable", values_to = "value") %>%
  ggplot(aes(x = value, y = GE)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  facet_wrap(~variable, scales = "free_x") +
  labs(x = "Continuous Variables", y = "Gastric Emptying t1/2 (minutes)")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# GE-Categorical
dat_long <- dat %>%
  pivot_longer(cols = all_of(categorical_vars), names_to = "variable", values_to = "group")
ggplot(dat_long, aes(x = factor(group), y = GE, fill = factor(group))) +
  geom_boxplot() +
  facet_wrap(~variable, scales = "free_x") +
  labs(x = "Group (0/1)", y = "Gastric Emptying t1/2 (minutes)",
       title = "GE by categorical variables") +
  theme(legend.position = "none")
```



The dataset contained very few missing values, which were removed without affecting the overall analysis.

Several continuous variables showed slight skewness, highlighting the need to check the normality of model residuals.

Sex was found to significantly influence Glucagon and PYY levels, suggesting it should be included as a covariate in modeling.

Strong correlations were observed between certain variables (e.g., BMI–BW, HbA1c–GlucoseFasting), indicating potential multicollinearity that should be controlled for during model construction.

Most biomarkers showed weak to moderate linear relationships with GE, supporting the use of a multivariable linear or generalized linear model to evaluate their combined effects. appropriately checked.

Model Fitting

First, we fitted all variables into the model, as model 1.

```
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

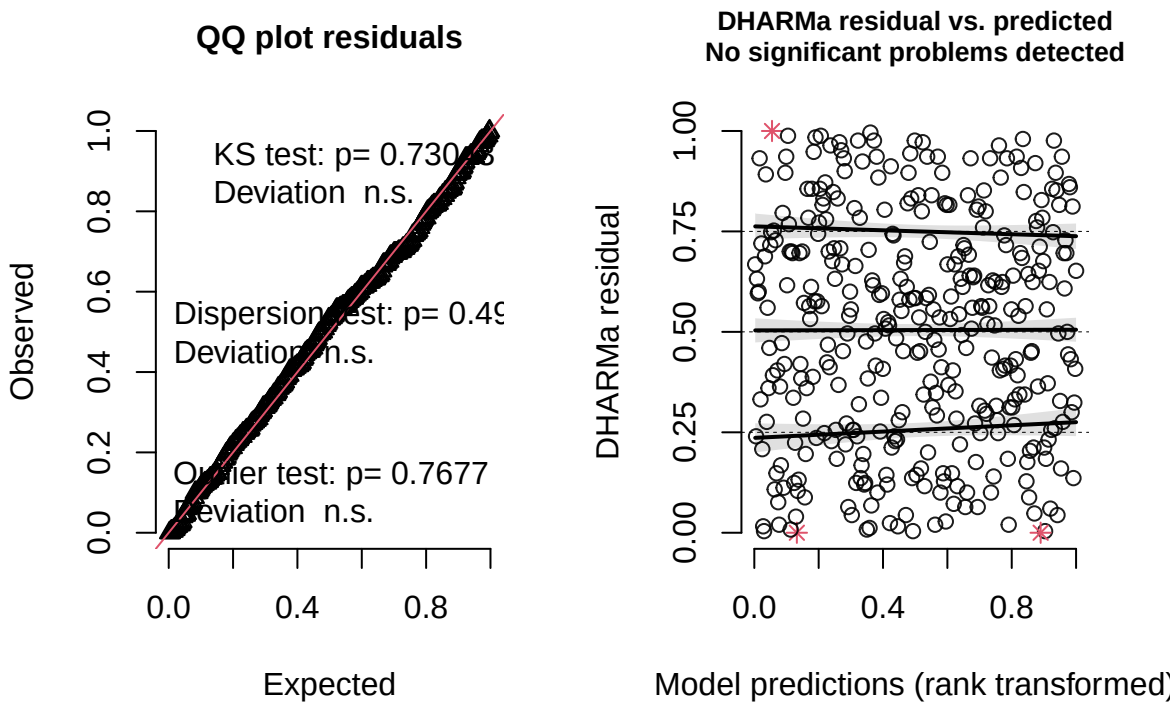
```r
library(DHARMa)
```

```
## This is DHARMa 0.4.7. For overview type '?DHARMa'. For recent changes, type news(package = 'DHARMa')
```

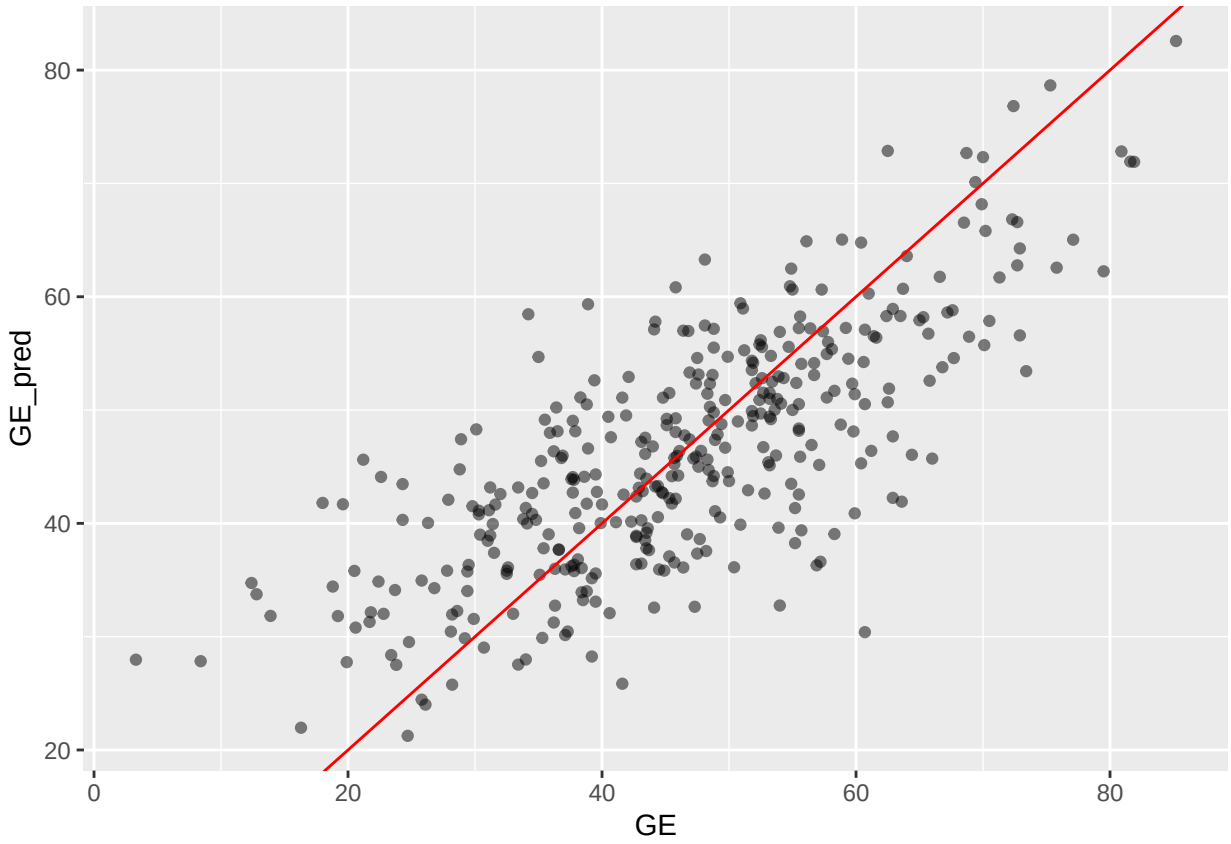
```r
library(interactions)
library(rsq)
model1 <- glm(GE ~ Sex + Age + Height + BW + BMI + GlucoseFasting + InsulinFasting + HbA1c + MatsudaIdx
              data = dat, family = gaussian)
summary(model1)
```

```
##
## Call:
## glm(formula = GE ~ Sex + Age + Height + BW + BMI + GlucoseFasting +
##     InsulinFasting + HbA1c + MatsudaIdx + HOMAB + DiabetesComplications +
##     Metformin + Gastrin + CCK + Ghrelin + Amylin + Glucagon +
##     GLP1 + PYY, family = gaussian, data = dat)
##
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           55.786887  59.176889   0.943  0.34649
## Sex                    3.276637   1.017970   3.219  0.00141 **
## Age                    0.077615   0.046367   1.674  0.09506 .
## Height                -0.287497   0.332842  -0.864  0.38832
## BW                     0.248418   0.357074   0.696  0.48708
## BMI                   -0.626067   1.744423  -0.359  0.71989
## GlucoseFasting        -0.649149   0.938698  -0.692  0.48969
## InsulinFasting        -0.124758   4.982047  -0.025  0.98004
## HbA1c                  0.372162   2.157608   0.172  0.86316
## MatsudaIdx            -0.827691   2.062192  -0.401  0.68840
## HOMAB                 -0.834878   7.335446  -0.114  0.90945
## DiabetesComplications  2.790453   2.102818   1.327  0.18539
## Metformin             -2.495386   1.127754  -2.213  0.02757 *
## Gastrin                0.061452   0.034619   1.775  0.07677 .
## CCK                          NA         NA      NA       NA
## Ghrelin                0.048553   0.004158  11.678  < 2e-16 ***
## Amylin                -0.082151   0.144171  -0.570  0.56917
## Glucagon               0.120889   0.174980   0.691  0.49011
## GLP1                   5.736636   0.342377  16.755  < 2e-16 ***
## PYY                    0.091999   0.048907   1.881  0.06080 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 88.92623)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30591  on 344  degrees of freedom
## AIC: 2679.7
##
## Number of Fisher Scoring iterations: 2
```

```r
simres1 <- simulateResiduals(model1)
plot(simres1)
```




```
dat$GE_pred <- predict(model1, newdata = dat, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```




```
rsq(model1, adj = TRUE)
```

```
## [1] 0.5483745
```

Sex is significantly associated with gastric emptying (GE) (p = 0.00141), indicating that males and females differ in GE.

Metformin usage shows a significant negative association with GE (p = 0.02757), suggesting slower GE in users.

Ghrelin and GLP1 have very strong positive effects on GE (p < 2e-16 for both), indicating higher levels correspond to longer GE half-life.

Gastrin and PYY show marginal associations (p = 0.06–0.077), suggesting a possible weak positive effect.

Age also shows a marginal effect (p = 0.095), with older age slightly increasing GE.

CCK is not defined due to singularity, which indicates perfect collinearity with another variable (likely Gastrin), so it was dropped from the model.

Other variables, including Height, BW, BMI, GlucoseFasting, InsulinFasting, HbA1c, MatsudaIdx, HOMAB, DiabetesComplications, Amylin, Glucagon, do not show significant associations with GE in this model.

```
model2 <- glm(GE ~ Sex + Age + Height + BW + BMI + GlucoseFasting + InsulinFasting + HbA1c + MatsudaIdx
              data = dat, family = gaussian)
summary(model2)
```

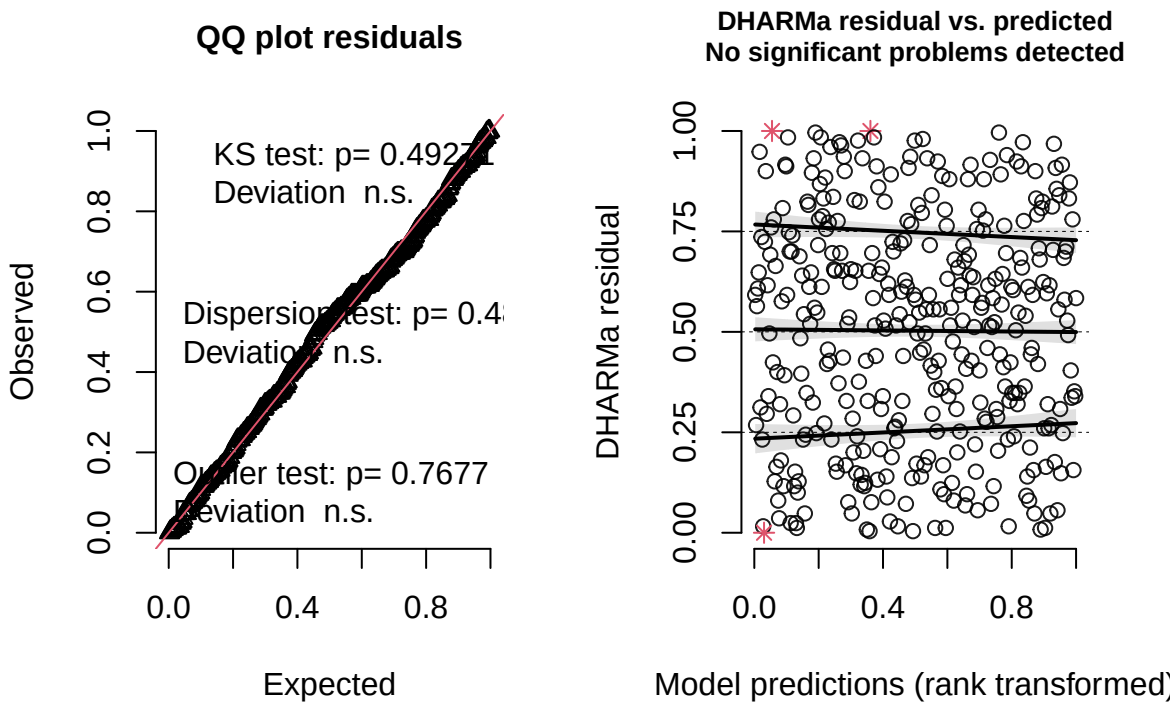
```
##
```

```
## Call:
## glm(formula = GE ~ Sex + Age + Height + BW + BMI + GlucoseFasting +
##     InsulinFasting + HbA1c + MatsudaIdx + HOMAB + DiabetesComplications +
##     Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1 +
##     PYY, family = gaussian, data = dat)
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           55.786887  59.176889   0.943  0.34649
## Sex                    3.276637   1.017970   3.219  0.00141 **
## Age                    0.077615   0.046367   1.674  0.09506 .
## Height                -0.287497   0.332842  -0.864  0.38832
## BW                     0.248418   0.357074   0.696  0.48708
## BMI                   -0.626067   1.744423  -0.359  0.71989
## GlucoseFasting        -0.649149   0.938698  -0.692  0.48969
## InsulinFasting        -0.124758   4.982047  -0.025  0.98004
## HbA1c                  0.372162   2.157608   0.172  0.86316
## MatsudaIdx            -0.827691   2.062192  -0.401  0.68840
## HOMAB                 -0.834878   7.335446  -0.114  0.90945
## DiabetesComplications  2.790453   2.102818   1.327  0.18539
## Metformin             -2.495386   1.127754  -2.213  0.02757 *
## Gastrin                0.061452   0.034619   1.775  0.07677 .
## Ghrelin                0.048553   0.004158  11.678  < 2e-16 ***
## Amylin                -0.082151   0.144171  -0.570  0.56917
## Glucagon               0.120889   0.174980   0.691  0.49011
## GLP1                   5.736636   0.342377  16.755  < 2e-16 ***
## PYY                    0.091999   0.048907   1.881  0.06080 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 88.92623)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30591  on 344  degrees of freedom
## AIC: 2679.7
##
## Number of Fisher Scoring iterations: 2
```

```
rsq(model2, adj = TRUE)
```

```
## [1] 0.5483745
```

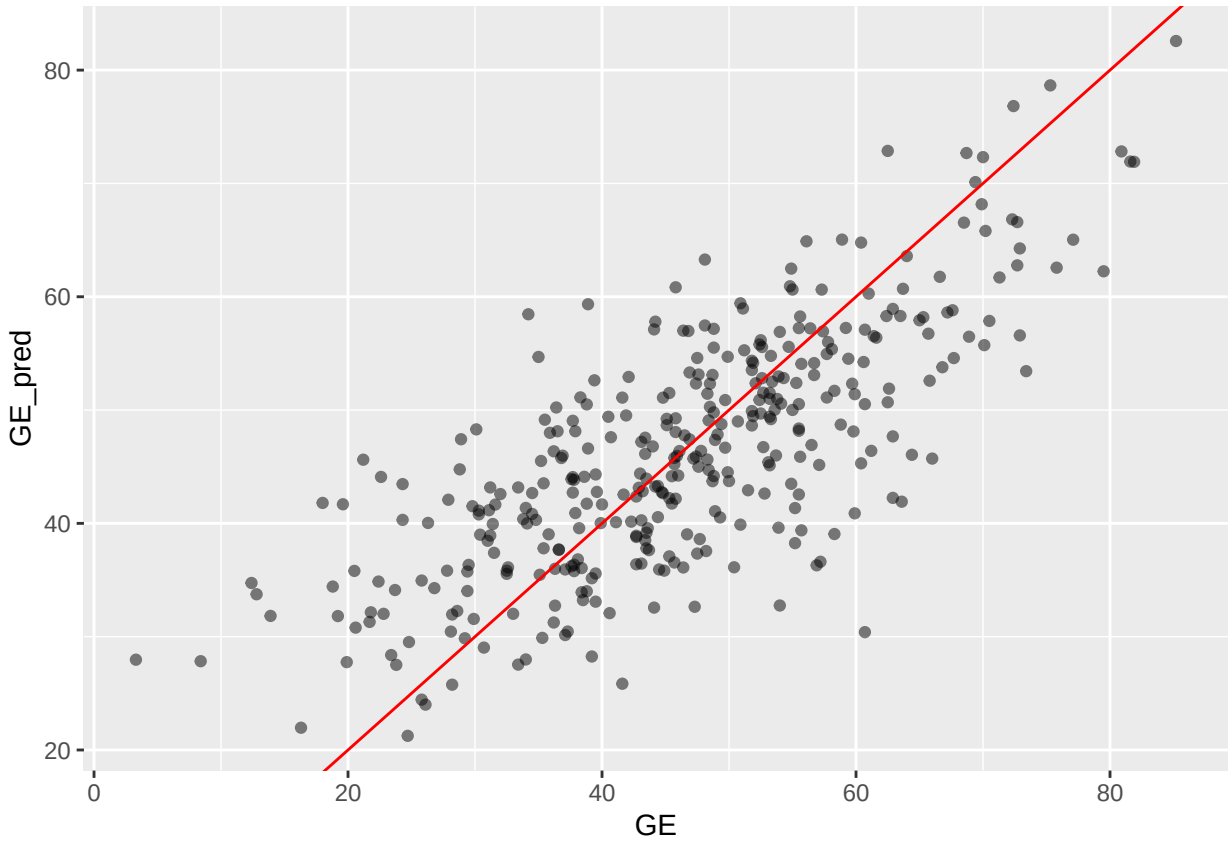
```
simres2 <- simulateResiduals(model2)
plot(simres2)
```




```
vif(model2)
```

```
##                  Sex                  Age               Height
##             1.056865             1.107838            41.884468
##                   BW                  BMI       GlucoseFasting
##            54.107466           284.271801            16.169906
##       InsulinFasting                HbA1c          MatsudaIdx
##           665.708971            16.224913            29.700250
##                HOMAB DiabetesComplications            Metformin
##           852.264193             1.071208             1.042593
##              Gastrin              Ghrelin               Amylin
##             1.046581             1.040432             1.033650
##             Glucagon                 GLP1                  PYY
##             1.047848             1.084886             1.069794
```

```
dat$GE_pred <- predict(model2, newdata = dat, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```




```
# examine model
# analyze_model <- function(model, data, pred_var=NULL, modx_var=NULL) {
#   print(summary(model))
#   print(rsq(model, adj = TRUE))
#
#   simres <- DHARMa::simulateResiduals(model)
#   plot(simres)
#
#   data$GE_pred <- predict(model, newdata = data, type = "response", na.action = na.pass)
#   ggplot(data, aes(x = GE, y = GE_pred)) +
#     geom_point(alpha = 0.5) +
#     geom_abline(intercept = 0, slope = 1, color = "red") +
#     ggtitle(deparse(substitute(model)))
#
#   if (!is.null(pred_var) & !is.null(modx_var)) {
#     interactions::interact_plot(model, pred = pred_var, modx = modx_var)
#   }
# }
# analyze_model(model1, dat[common_rows, ])
# analyze_model(model2, dat[common_rows, ])
#
# compare_models <- function(model1, model2) {
#   print(AIC(model1))
#   print(AIC(model2))
#   print(BIC(model1))
#   print(BIC(model2))
#   print(anova(model1, model2, test = "Chisq"))
```

```
# }
# compare_models(model2, model3)
# models
```

When deleted CCK, the model remained exactly the same.

Moreover, we can see that Height, BW, BMI, GlucoseFasting, InsulinFasting, HOMAB have significantly high VIF, indicating colinearality. MatsudaIdx, HbA1c, GlucoseFasting also showed relatively high VIF.

Because BMI contains information about both height and BodyWeight, we delete height and BW in the following model. HOMA-B is an index used to estimate pancreatic β -cell function based on fasting plasma glucose and fasting insulin levels. Thus, we delete InsulinFasting and only keep HOMAB. GlucoseFasting refers to the blood glucose concentration measured after at least 8 hours of fasting and is an important indicator for assessing glucose metabolism and diagnosing diabetes. HbA1c is a product of non-enzymatic glycation of hemoglobin by glucose in the blood, reflecting the average blood glucose level over the past 2-3 months. It is an important indicator for assessing diabetes control and long-term glycemic management. We chose Hb1Ac and deleted GlucoseFasting.

```
model3 <- glm(GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB +
                DiabetesComplications + Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1 + PYY,
              data = dat, family = gaussian)
summary(model3)
```

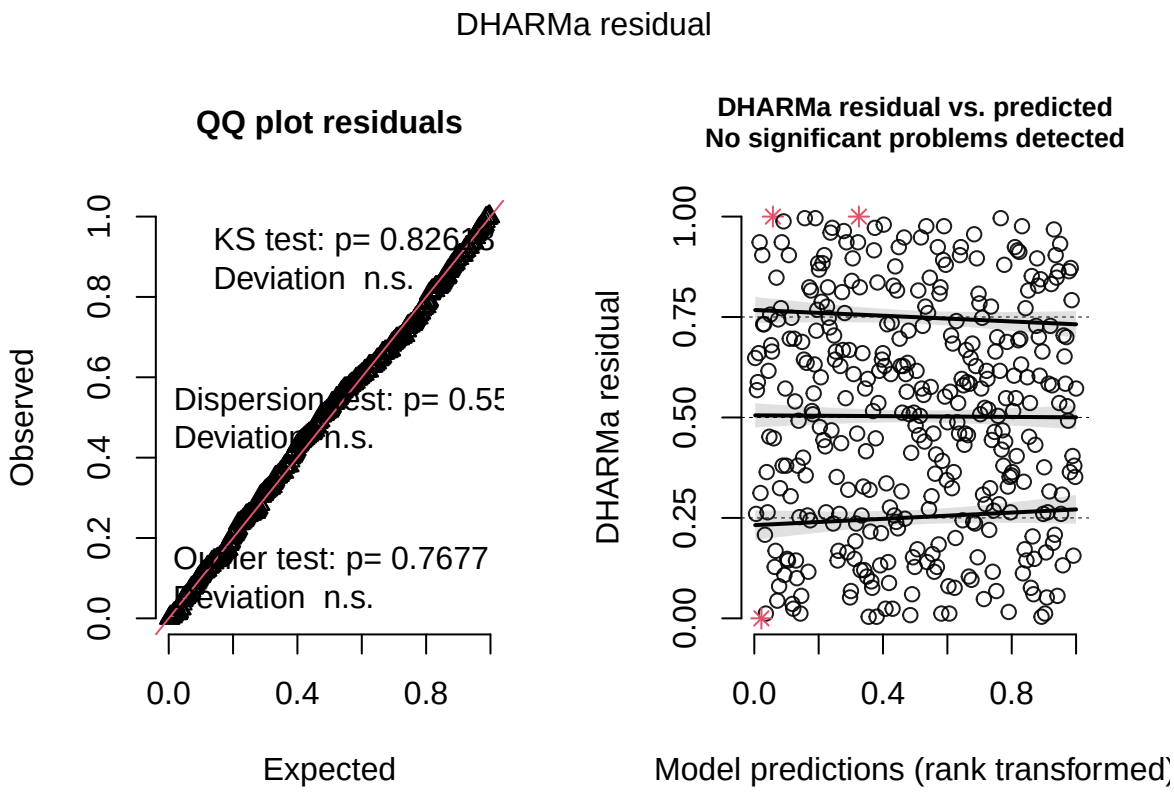
```
##
## Call:
## glm(formula = GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB +
##     DiabetesComplications + Metformin + Gastrin + Ghrelin + Amylin +
##     Glucagon + GLP1 + PYY, family = gaussian, data = dat)
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           11.456380  19.585200   0.585  0.55896
## Sex                    3.168883   1.009710   3.138  0.00184 **
## Age                    0.078960   0.044737   1.765  0.07844 .
## BMI                    0.188169   0.304308   0.618  0.53675
## HbA1c                 -1.081421   0.544892  -1.985  0.04797 *
## MatsudaIdx            -0.872763   2.017074  -0.433  0.66551
## HOMAB                 -1.021323   1.528502  -0.668  0.50446
## DiabetesComplications  2.948125   2.078970   1.418  0.15707
## Metformin             -2.358375   1.112575  -2.120  0.03473 *
## Gastrin                0.060547   0.034420   1.759  0.07944 .
## Ghrelin                0.047948   0.004107  11.674  < 2e-16 ***
## Amylin                -0.087973   0.143099  -0.615  0.53911
## Glucagon               0.144054   0.173282   0.831  0.40636
## GLP1                   5.724822   0.337153  16.980  < 2e-16 ***
## PYY                    0.092356   0.048539   1.903  0.05790 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 88.31691)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30734  on 348  degrees of freedom
## AIC: 2673.4
```

```
##
## Number of Fisher Scoring iterations: 2
```

```
rsq(model3, adj = TRUE)
```

```
## [1] 0.551469
```

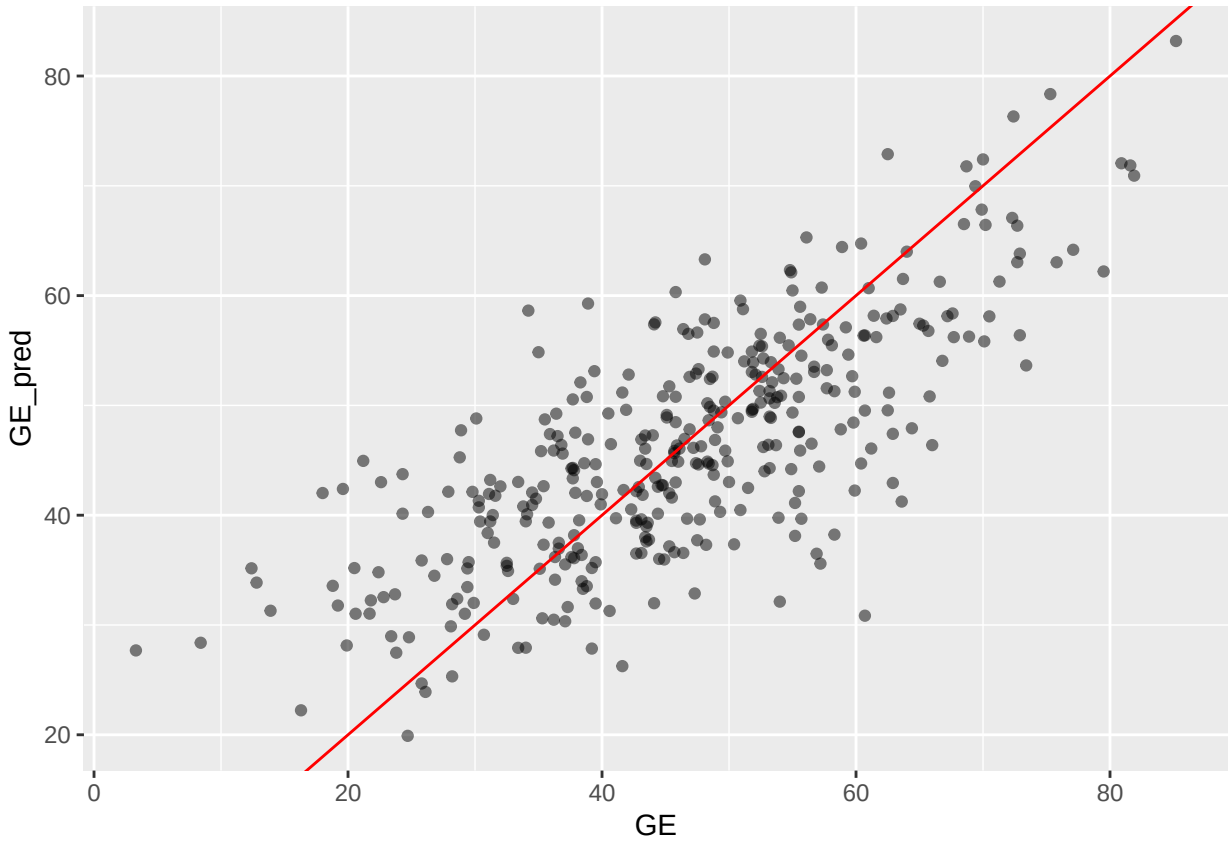
```
simres3 <- simulateResiduals(model3)
plot(simres3)
```




```
vif(model3)
```

```
##                    Sex                    Age                    BMI
##               1.046957               1.038397               8.710514
##                  HbA1c             MatsudaIdx                  HOMAB
##               1.041944              28.610909              37.259663
## DiabetesComplications              Metformin                Gastrin
##               1.054272               1.021717               1.041674
##                Ghrelin                 Amylin               Glucagon
##               1.022247               1.025364               1.034700
##                   GLP1                    PYY
##               1.059292               1.061018
```

```
dat$GE_pred <- predict(model3, newdata = dat, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```



The AIC value indicates that model3 has a slight improve on the model. Though Residual deviance rised slightly, the model is more stable. However, MatsudaIdx and HOMAB still has high VIF. The two variable represent different biological matters, so we look for ways to keep them both while dealing with the colinearality.

Standardization can improve coefficient stability. After standardization, variables are on a similar scale and numerical range, which makes the algorithm more stable when calculating the inverse matrix or least squares, reducing the impact of multicollinearity.

```
dat_scaled <- dat
dat_scaled[all_vars] <- scale(dat[all_vars])

model4 <- glm(GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB +
                DiabetesComplications + Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1 + PYY,
              data = dat_scaled, family = gaussian)
summary(model4)
```

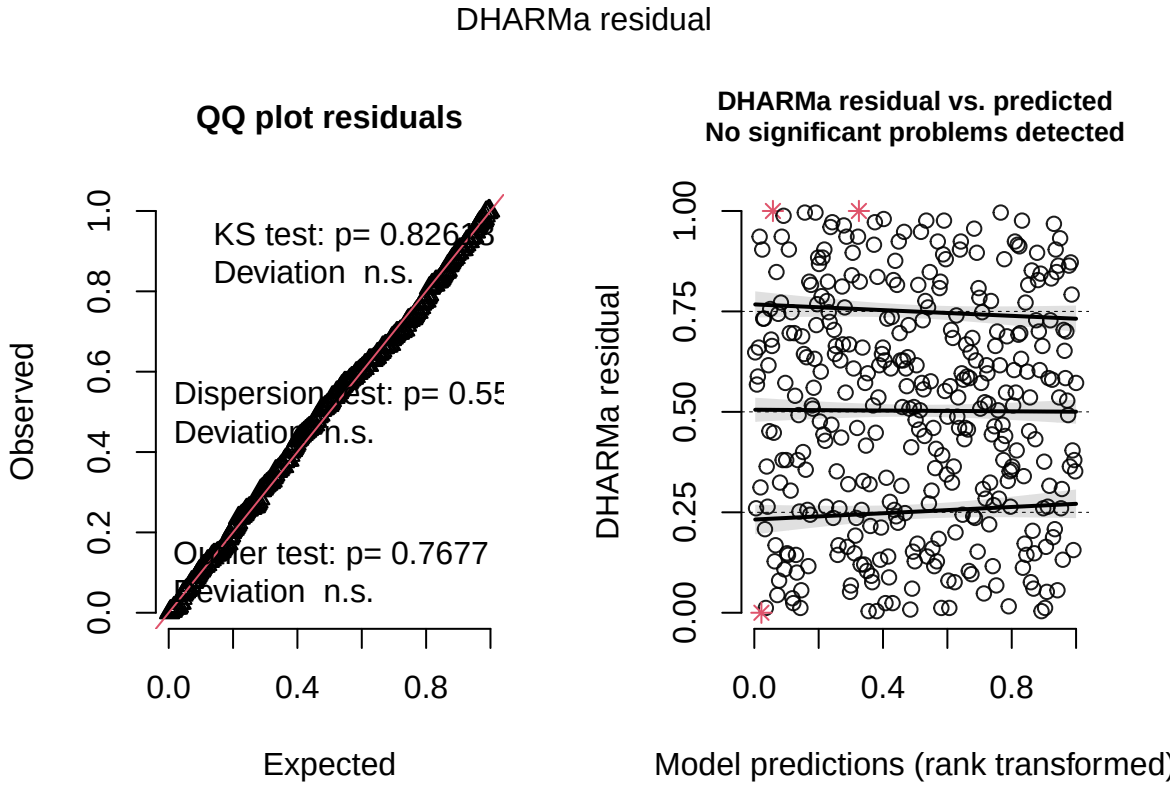
```
##
## Call:
## glm(formula = GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB +
##     DiabetesComplications + Metformin + Gastrin + Ghrelin + Amylin +
##     Glucagon + GLP1 + PYY, family = gaussian, data = dat_scaled)
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.1881     0.7741  58.376  < 2e-16 ***
## Sex                     3.1689     1.0097   3.138  0.00184 **
## Age                     0.8884     0.5033   1.765  0.07844 .
```

```
## BMI                       0.9014     1.4578   0.618  0.53675
## HbA1c                    -1.0006     0.5042  -1.985  0.04797 *
## MatsudaIdx               -1.1432     2.6420  -0.433  0.66551
## HOMAB                    -2.0146     3.0150  -0.668  0.50446
## DiabetesComplications     2.9481     2.0790   1.418  0.15707
## Metformin                -2.3584     1.1126  -2.120  0.03473 *
## Gastrin                   0.8868     0.5041   1.759  0.07944 .
## Ghrelin                   5.8302     0.4994  11.674  < 2e-16 ***
## Amylin                   -0.3075     0.5002  -0.615  0.53911
## Glucagon                  0.4177     0.5024   0.831  0.40636
## GLP1                      8.6320     0.5084  16.980  < 2e-16 ***
## PYY                       0.9681     0.5088   1.903  0.05790 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 88.31691)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30734  on 348  degrees of freedom
## AIC: 2673.4
##
## Number of Fisher Scoring iterations: 2
```

```
rsq(model4, adj = TRUE)
```

```
## [1] 0.551469
```

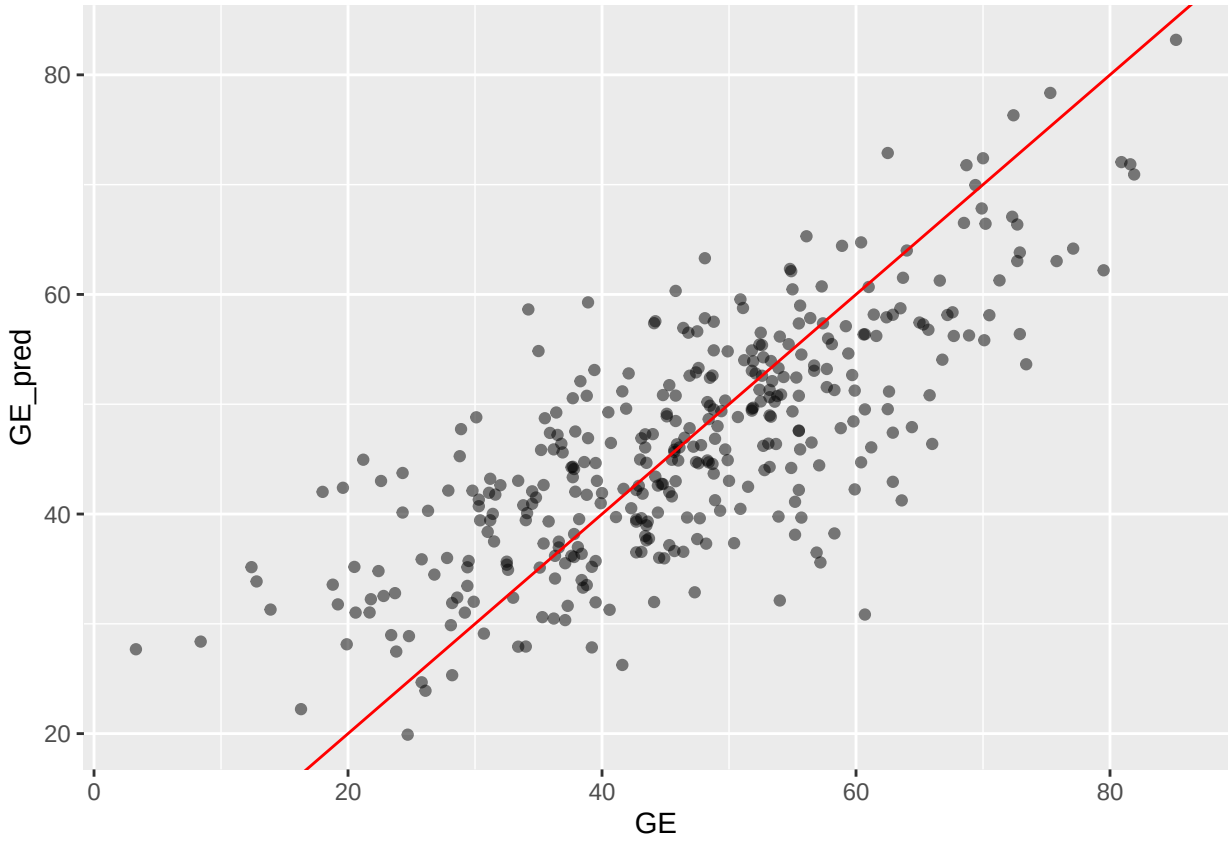
```
simres4 <- simulateResiduals(model4)
plot(simres4)
```



```r
vif(model4)
```

```
##                    Sex                    Age                    BMI
##               1.046957               1.038397               8.710514
##                  HbA1c             MatsudaIdx                  HOMAB
##               1.041944              28.610909              37.259663
## DiabetesComplications              Metformin                Gastrin
##               1.054272               1.021717               1.041674
##                Ghrelin                 Amylin               Glucagon
##               1.022247               1.025364               1.034700
##                   GLP1                    PYY
##               1.059292               1.061018
```

```r
dat$GE_pred <- predict(model4, newdata = dat_scaled, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```



After Standerdization, the parameter changed slightly.

Next, we add covariates to the model, according to early data exploration. A covariate is a variable that is potentially related to both the dependent variable (outcome) and one or more independent variables (predictors) in a statistical model.Including covariates allows us to control for confounding, reduce error variance, and obtain a more accurate estimate of the main effect of interest.

We first try to time PYY with Sex, since they both have significant impact on the model.

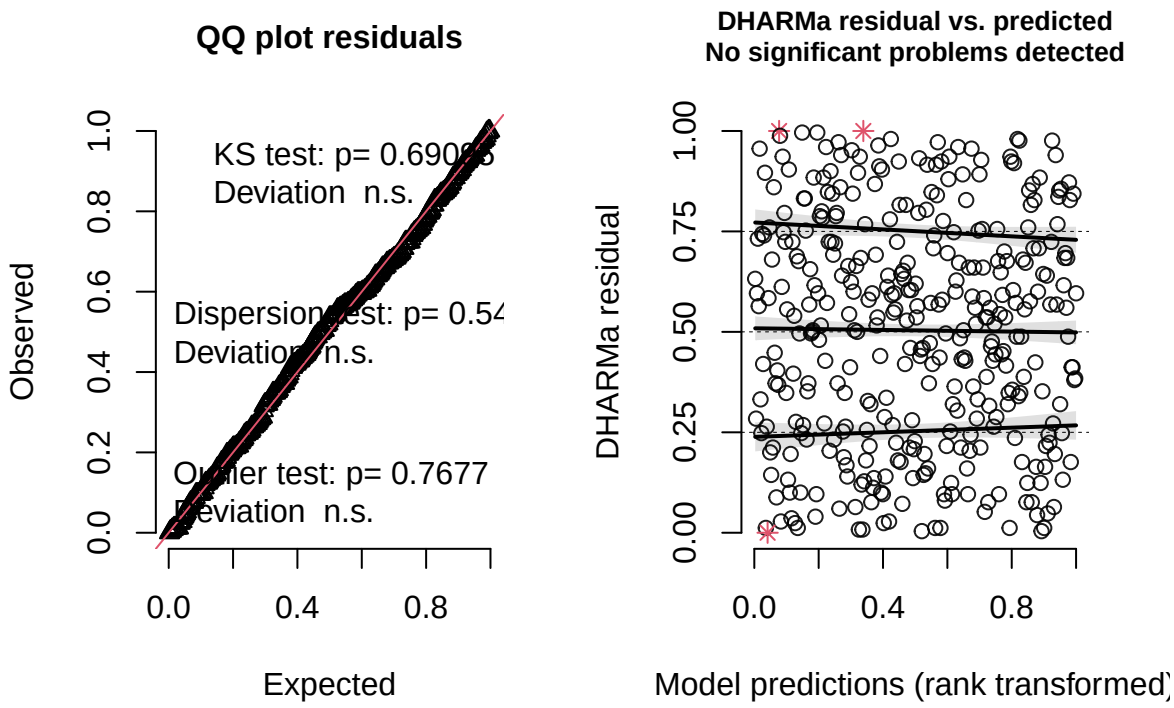
```r
# model
model5 <- glm(GE ~ Sex * PYY + Age + BMI + HbA1c + MatsudaIdx + HOMAB +
                DiabetesComplications + Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1,
              data = dat_scaled, family = gaussian)
summary(model5)
```

```
##
## Call:
## glm(formula = GE ~ Sex * PYY + Age + BMI + HbA1c + MatsudaIdx +
##     HOMAB + DiabetesComplications + Metformin + Gastrin + Ghrelin +
##     Amylin + Glucagon + GLP1, family = gaussian, data = dat_scaled)
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              45.1077     0.7732  58.340  < 2e-16 ***
## Sex                       3.1610     1.0068   3.140  0.00184 **
## PYY                       1.8949     0.7339   2.582  0.01023 *
## Age                       0.8451     0.5025   1.682  0.09349 .
## BMI                       0.5733     1.4656   0.391  0.69591
## HbA1c                    -0.9721     0.5030  -1.933  0.05409 .
## MatsudaIdx               -0.6303     2.6505  -0.238  0.81218
## HOMAB                    -1.4152     3.0256  -0.468  0.64026
## DiabetesComplications     2.6135     2.0817   1.255  0.21015
## Metformin                -2.3474     1.1093  -2.116  0.03505 *
## Gastrin                   0.8577     0.5029   1.706  0.08899 .
## Ghrelin                   5.8821     0.4988  11.792  < 2e-16 ***
## Amylin                   -0.3060     0.4987  -0.614  0.53990
## Glucagon                  0.4297     0.5010   0.858  0.39162
## GLP1                      8.6253     0.5069  17.016  < 2e-16 ***
## Sex:PYY                  -1.7622     1.0084  -1.748  0.08142 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 87.7987)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30466  on 347  degrees of freedom
## AIC: 2672.2
##
## Number of Fisher Scoring iterations: 2
```

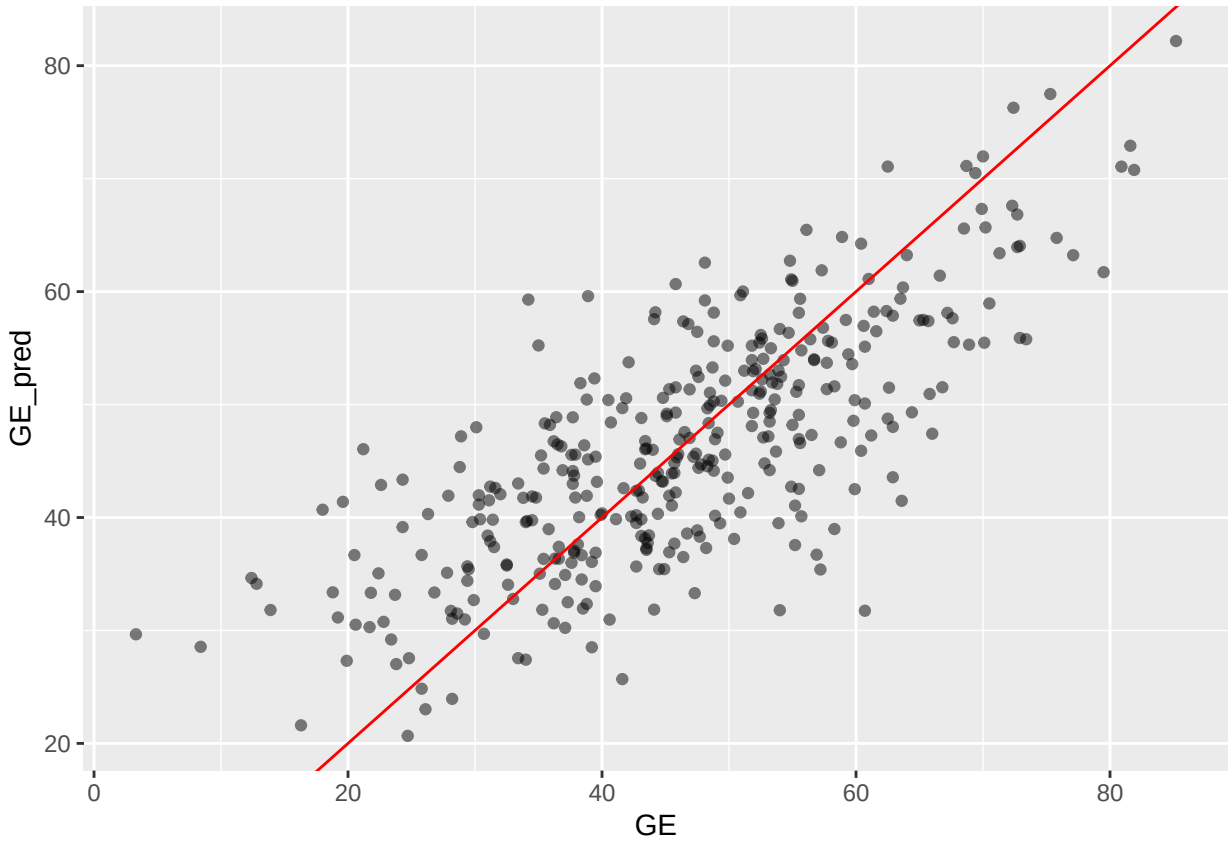
```r
rsq(model5, adj = TRUE)
```

```
## [1] 0.5541008
```

```r
simres5 <- simulateResiduals(model5)
plot(simres5)
```




```r
dat$GE_pred <- predict(model5, newdata = dat_scaled, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```



The results showed that model5 has a lower AIC and Residual deviance, which indicate a higher performance.

The interaction term Sex:PYY (β = -1.76, p = 0.081) shows a trend-level effect, meaning that it is not statistically significant at the conventional 0.05 level, but it approaches significance.In females (reference group), PYY has a positive association with GE (β = +1.89, p = 0.010), indicating that higher PYY levels are related to faster or greater gastric emptying. In males, the total effect of PYY becomes much weaker, suggesting that the positive relationship between PYY and GE observed in females is largely diminished or absent in males. Thus, the trend-level interaction implies that sex may modulate the physiological effect of PYY, but the evidence is not strong enough to claim a definitive moderating effect.

We move on to comparing model5 with model4.

```
# examine
AIC(model4, model5)
```

```
##        df      AIC
## model4 16 2673.409
## model5 17 2672.228
```

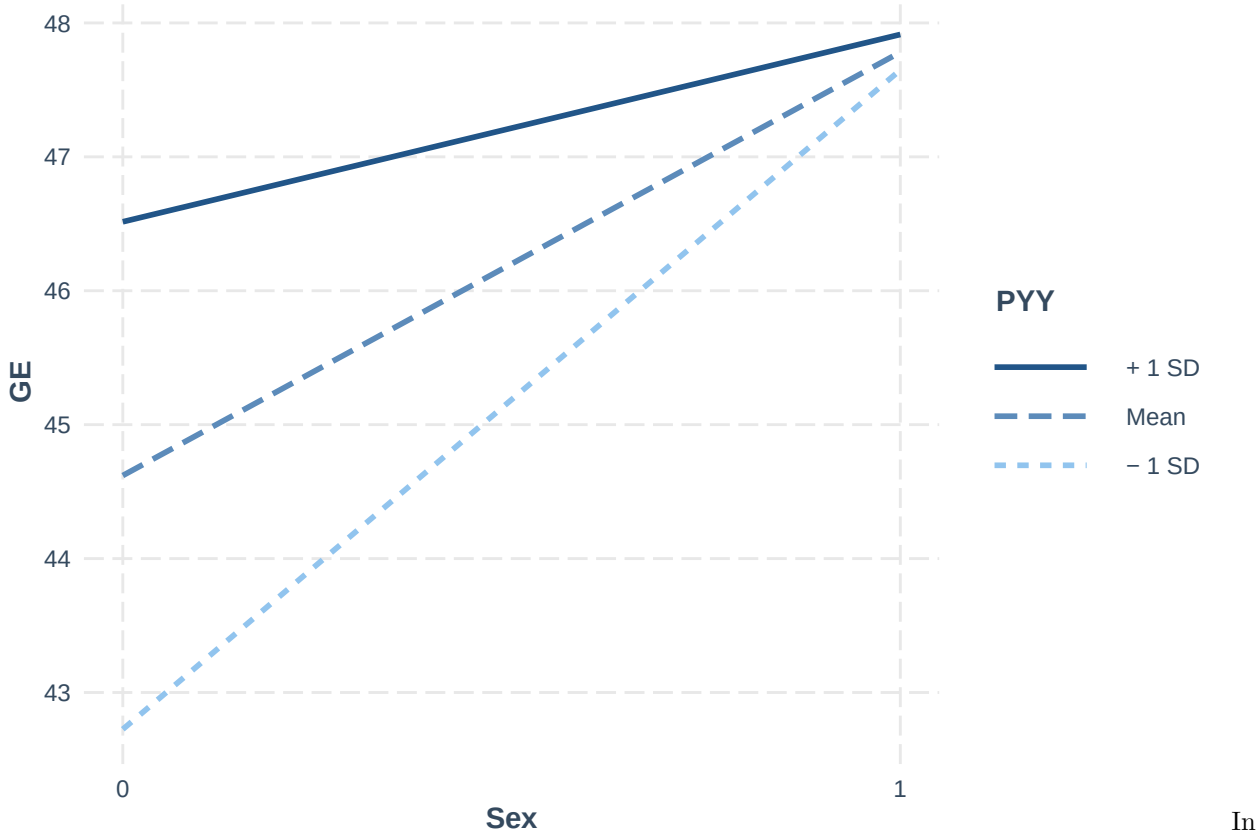
```
BIC(model4, model5)
```

```
##        df      BIC
## model4 16 2735.719
## model5 17 2738.433
```

```
anova(model4, model5, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB + DiabetesComplications +
##     Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1 +
##     PYY
## Model 2: GE ~ Sex * PYY + Age + BMI + HbA1c + MatsudaIdx + HOMAB + DiabetesComplications +
##     Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       348      30734
## 2       347      30466  1   268.14  0.08054 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
library(interactions)
interact_plot(model5, pred = "Sex", modx = "PYY")
```



In regression models, an interaction term such as A × B represents the idea that the effect of variable A on the outcome depends on variable B, and vice versa.

However, if one of these variables has no main effect - meaning it does not show any significant relationship with the dependent variable - then adding an interaction term is often statistically and conceptually unstable.

But in this work we are going to try and see the result of these interaction terms. Metformin has a significant impact, but HOMAB doesn't.

```
# model
model6 <- glm(GE ~ Metformin * HOMAB + Sex + PYY + Age + BMI + HbA1c + MatsudaIdx +
                DiabetesComplications + Gastrin + Ghrelin + Amylin + Glucagon + GLP1,
```

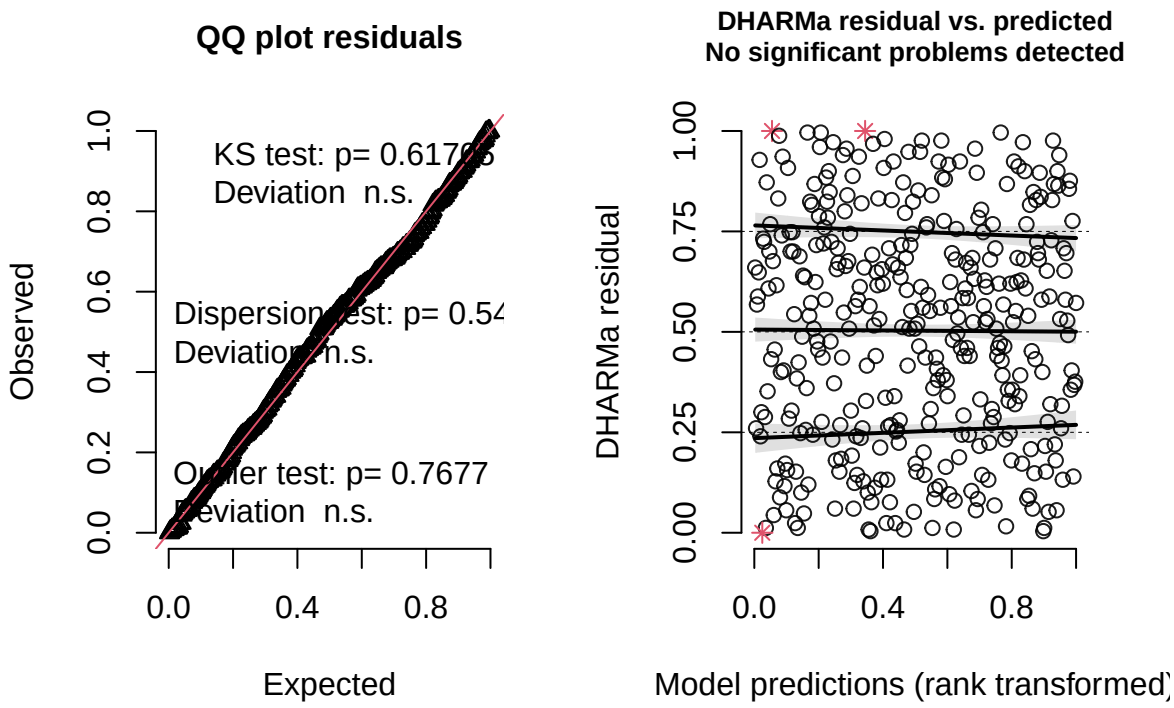
```
              data = dat_scaled, family = gaussian)
summary(model6)
```

```
##
## Call:
## glm(formula = GE ~ Metformin * HOMAB + Sex + PYY + Age + BMI +
##     HbA1c + MatsudaIdx + DiabetesComplications + Gastrin + Ghrelin +
##     Amylin + Glucagon + GLP1, family = gaussian, data = dat_scaled)
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            45.2270     0.7756  58.314  < 2e-16 ***
## Metformin              -2.2498     1.1197  -2.009  0.04528 *
## HOMAB                  -2.1899     3.0225  -0.725  0.46923
## Sex                     3.1256     1.0112   3.091  0.00216 **
## PYY                     0.9697     0.5089   1.905  0.05757 .
## Age                     0.8704     0.5039   1.727  0.08499 .
## BMI                     0.8506     1.4594   0.583  0.56036
## HbA1c                  -0.9616     0.5063  -1.899  0.05835 .
## MatsudaIdx             -1.0979     2.6433  -0.415  0.67815
## DiabetesComplications   2.8834     2.0809   1.386  0.16675
## Gastrin                 0.9113     0.5050   1.804  0.07205 .
## Ghrelin                 5.8303     0.4996  11.671  < 2e-16 ***
## Amylin                 -0.2704     0.5021  -0.539  0.59055
## Glucagon                0.3870     0.5038   0.768  0.44284
## GLP1                    8.6377     0.5086  16.984  < 2e-16 ***
## Metformin:HOMAB         1.0707     1.2117   0.884  0.37753
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 88.37259)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30665  on 347  degrees of freedom
## AIC: 2674.6
##
## Number of Fisher Scoring iterations: 2
```

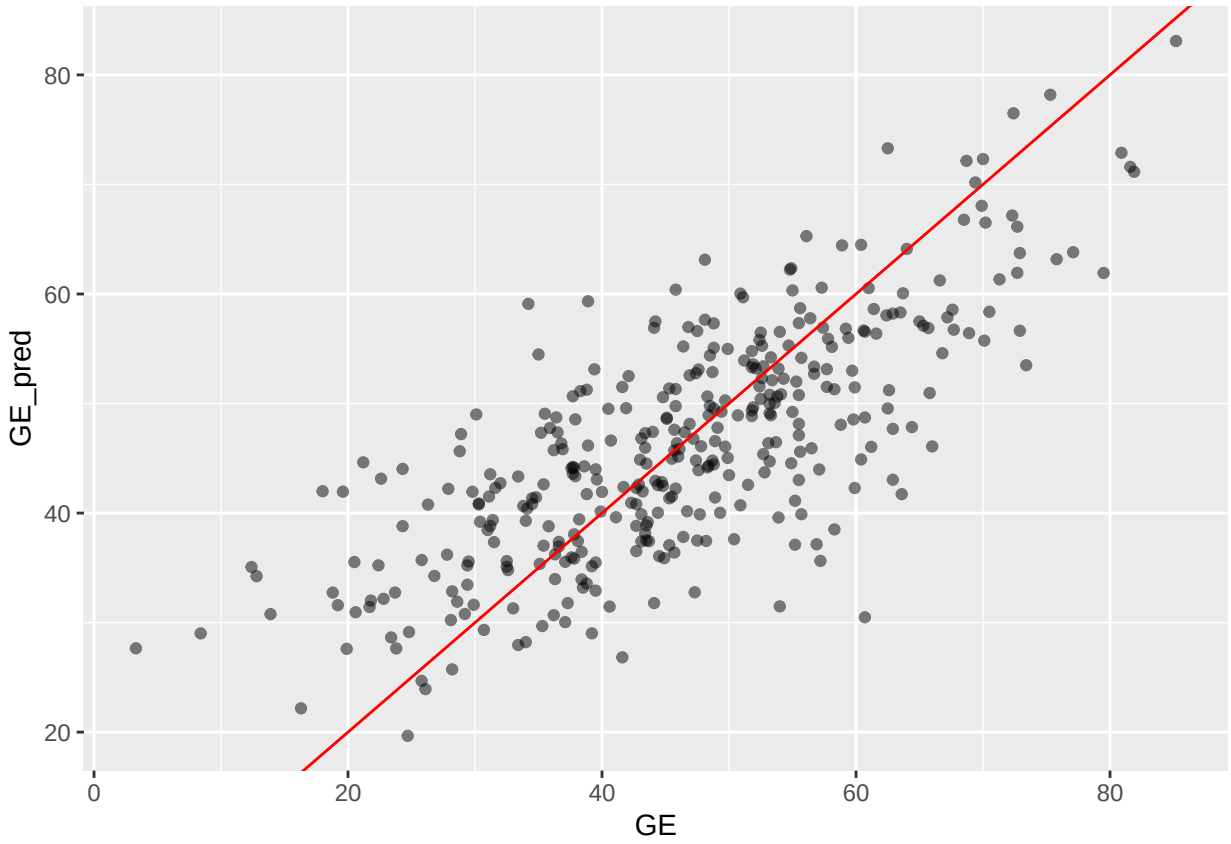
```
rsq(model6, adj = TRUE)
```

```
## [1] 0.5511862
```

```
simres6 <- simulateResiduals(model6)
plot(simres6)
```




```r
dat$GE_pred <- predict(model6, newdata = dat_scaled, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```



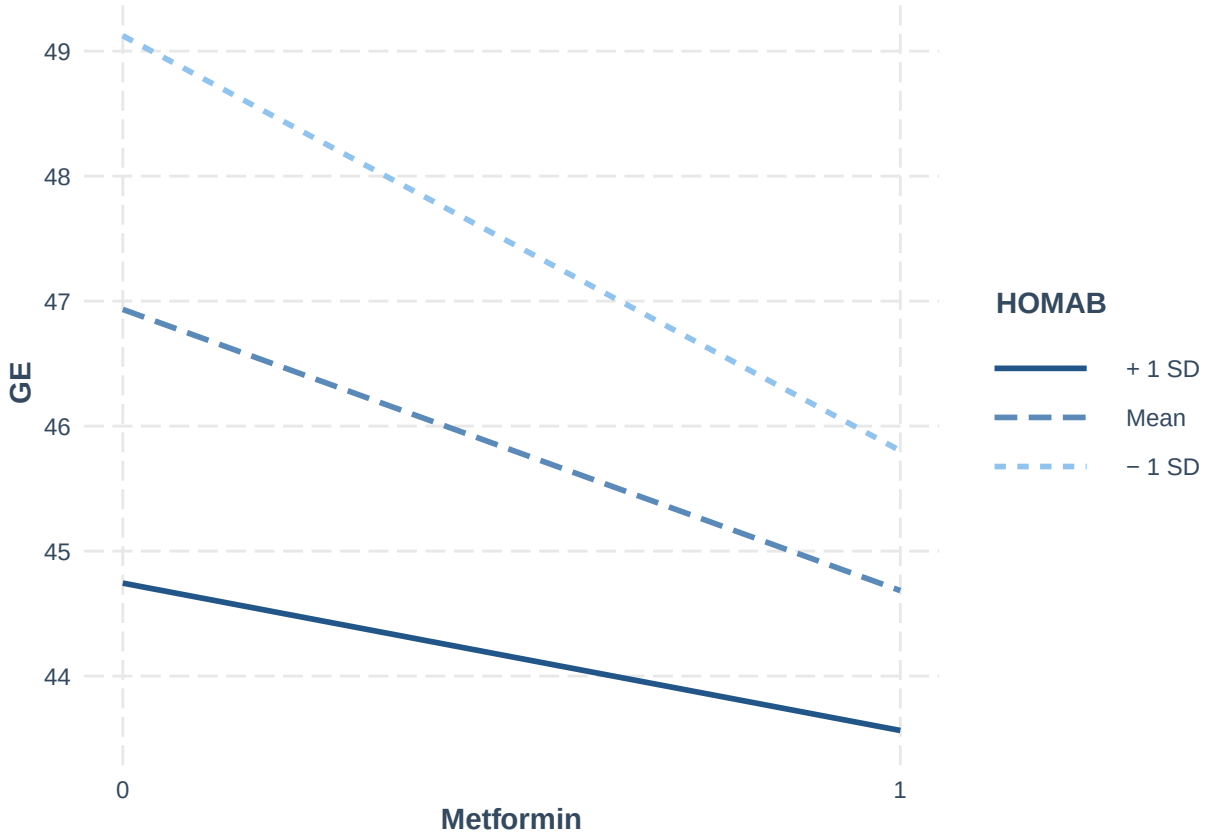

```
AIC(model4, model6)
```

```
##        df      AIC
## model4 16 2673.409
## model6 17 2674.593
```

```
BIC(model4, model6)
```

```
##        df      BIC
## model4 16 2735.719
## model6 17 2740.798
```

```
interact_plot(model6, pred = "Metformin", modx = "HOMAB")
```



From the result we can see that the interaction term did not work on improving the model.

Let's try DiabetesComplications-GLP1, in which DiabetesComplications does not have significant impact and GLP1 does.

```
# model
model7 <- glm(GE ~ DiabetesComplications * GLP1 + Metformin + HOMAB + Sex + PYY + Age + BMI + HbA1c +
                MatsudaIdx +Gastrin + Ghrelin + Amylin + Glucagon,
              data = dat_scaled, family = gaussian)
summary(model7)
```

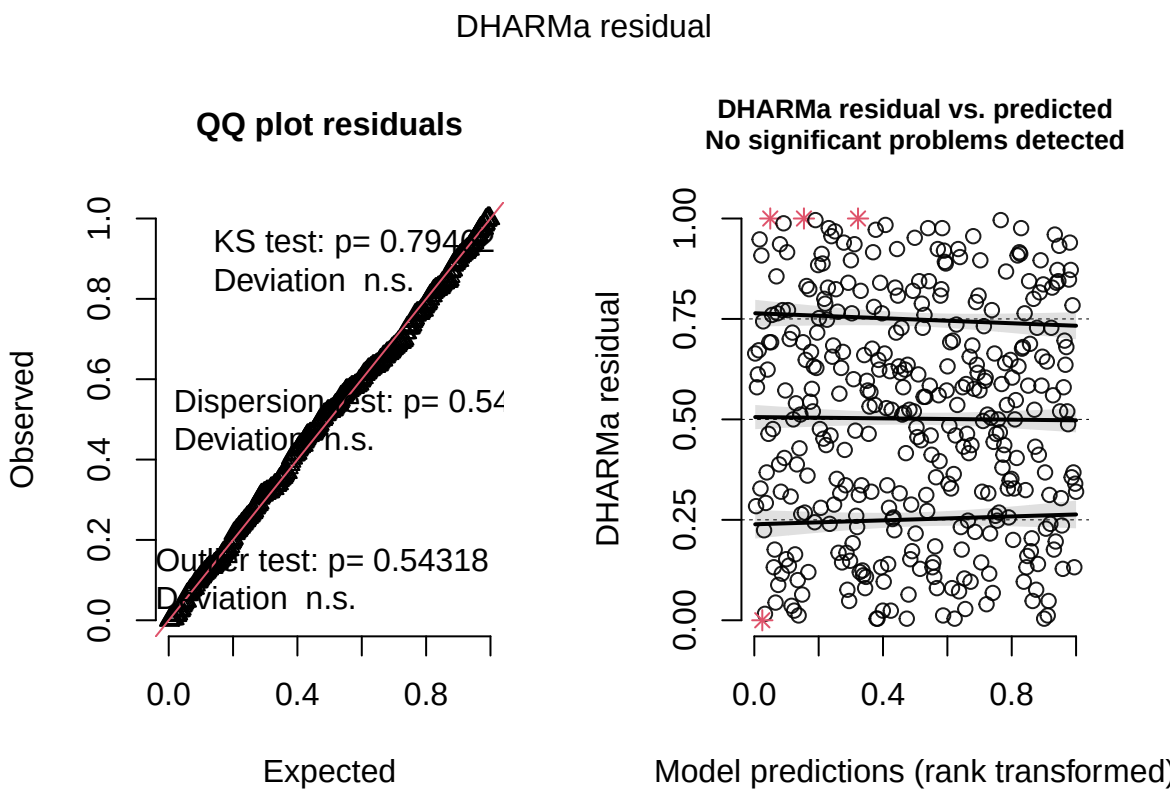
```
##
## Call:
## glm(formula = GE ~ DiabetesComplications * GLP1 + Metformin +
##     HOMAB + Sex + PYY + Age + BMI + HbA1c + MatsudaIdx + Gastrin +
##     Ghrelin + Amylin + Glucagon, family = gaussian, data = dat_scaled)
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  45.1364     0.7699  58.629  < 2e-16 ***
## DiabetesComplications         1.6076     2.1497   0.748  0.45507
## GLP1                          8.9438     0.5238  17.076  < 2e-16 ***
## Metformin                    -2.1906     1.1085  -1.976  0.04893 *
## HOMAB                        -1.9431     2.9974  -0.648  0.51725
## Sex                           3.1653     1.0038   3.153  0.00175 **
## PYY                           0.9034     0.5066   1.783  0.07541 .
## Age                           0.8527     0.5006   1.703  0.08941 .
## BMI                           0.9198     1.4492   0.635  0.52604
```

```
## HbA1c                        -1.0214     0.5013  -2.038  0.04235 *
## MatsudaIdx                   -1.0384     2.6268  -0.395  0.69287
## Gastrin                       0.8686     0.5012   1.733  0.08399 .
## Ghrelin                       5.8836     0.4970  11.838  < 2e-16 ***
## Amylin                       -0.2793     0.4974  -0.562  0.57476
## Glucagon                      0.4654     0.4999   0.931  0.35252
## DiabetesComplications:GLP1   -4.4361     1.9571  -2.267  0.02403 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 87.27915)
##
##     Null deviance: 71279  on 362  degrees of freedom
## Residual deviance: 30286  on 347  degrees of freedom
## AIC: 2670.1
##
## Number of Fisher Scoring iterations: 2
```

```
rsq(model7, adj = TRUE)
```

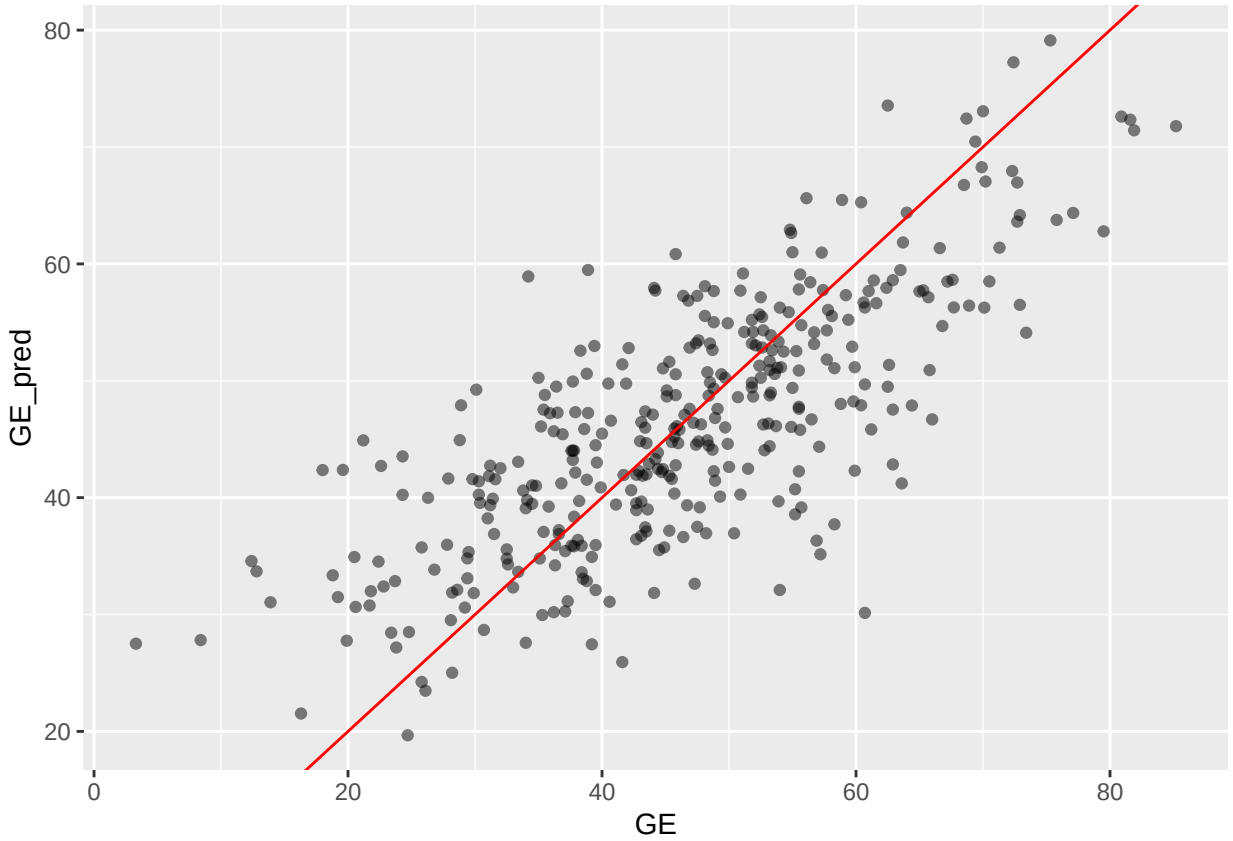
```
## [1] 0.5567395
```

```
simres7 <- simulateResiduals(model7)
plot(simres7)
```




```
dat$GE_pred <- predict(model7, newdata = dat_scaled, type = "response", na.action = na.pass)
ggplot(dat, aes(x = GE, y = GE_pred)) +
```

```
  geom_point(alpha = 0.5) +
  geom_abline(intercept = 0, slope = 1, color = "red")
```



Although DiabetesComplications itself is not a significant predictor of GE (p = 0.455), the interaction term DiabetesComplications × GLP1 is significant (p = 0.024).This indicates a moderation effect - the influence of GLP1 on GE differs depending on the presence of diabetes complications.

In individuals without complications, GLP1 has a strong positive association with GE ($\beta \approx 8.94$).

In individuals with complications, this effect is weakened ($\beta \approx 8.94$ - 4.44 ≈ 4.50).

Thus, GLP1's positive effect on GE is moderated by diabetes complications, suggesting that the physiological role of GLP1 may be partially impaired in patients with complications.

```
# examine
AIC(model4, model7)
```

```
##        df      AIC
## model4 16 2673.409
## model7 17 2670.074
```

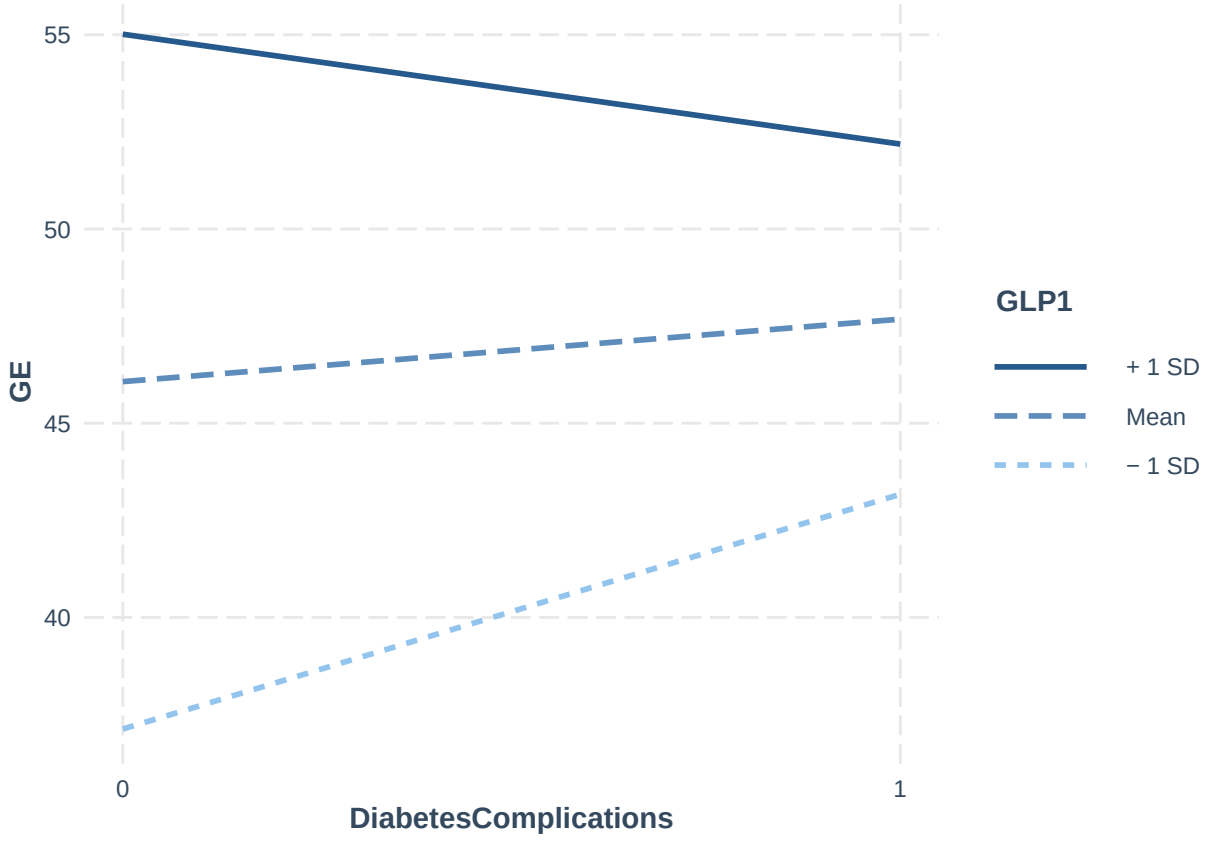
```
BIC(model4, model7)
```

```
##        df      BIC
## model4 16 2735.719
## model7 17 2736.278
```

```r
anova(model4, model7, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: GE ~ Sex + Age + BMI + HbA1c + MatsudaIdx + HOMAB + DiabetesComplications +
##     Metformin + Gastrin + Ghrelin + Amylin + Glucagon + GLP1 +
##     PYY
## Model 2: GE ~ DiabetesComplications * GLP1 + Metformin + HOMAB + Sex +
##     PYY + Age + BMI + HbA1c + MatsudaIdx + Gastrin + Ghrelin +
##     Amylin + Glucagon
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       348      30734
## 2       347      30286  1   448.42  0.02341 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
interact_plot(model7, pred = "DiabetesComplications", modx = "GLP1")
```




```r
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 4.1-10
```

```
x <- model.matrix(GE ~ DiabetesComplications + GLP1 + Metformin + HOMAB + Sex + PYY + Age + BMI + HbA1c
                  data = dat_scaled)[, -1]
y <- dat_scaled$GE
lasso_fit <- cv.glmnet(x, y, alpha = 1, family = "gaussian")

lasso_fit$lambda.min
```

```
## [1] 0.04580325
```

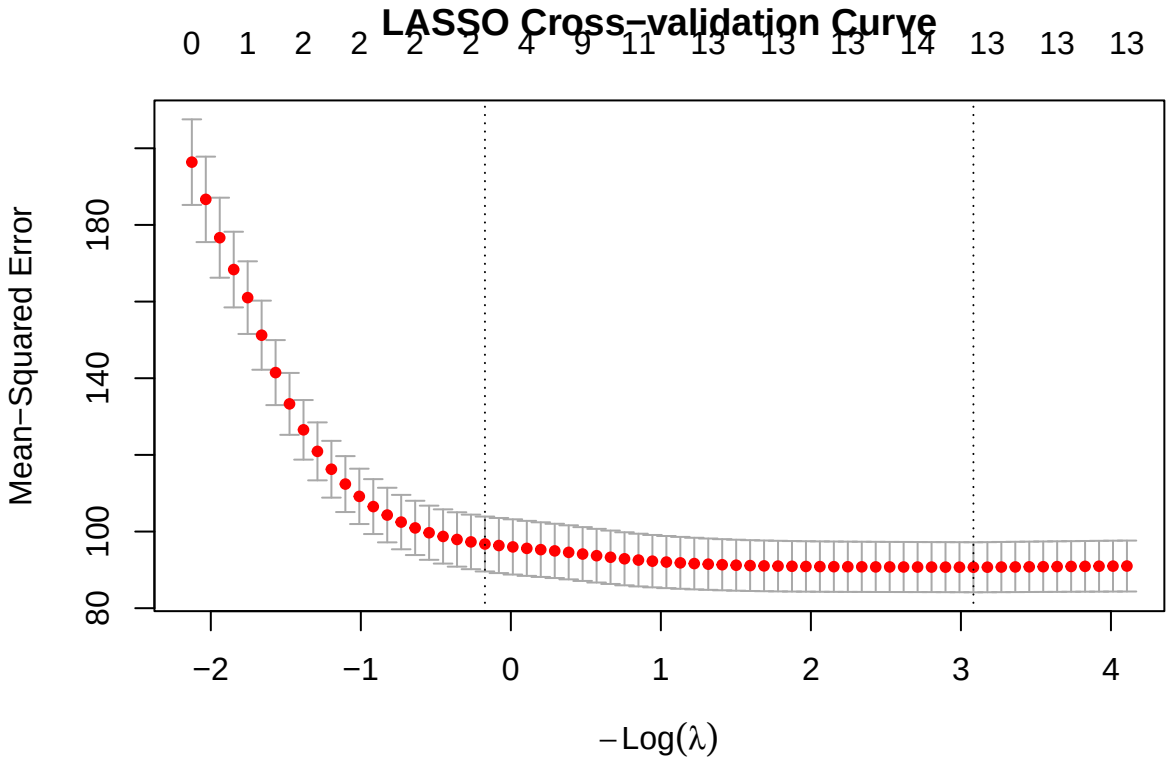
```
lasso_fit$lambda.1se
```

```
## [1] 1.188606
```

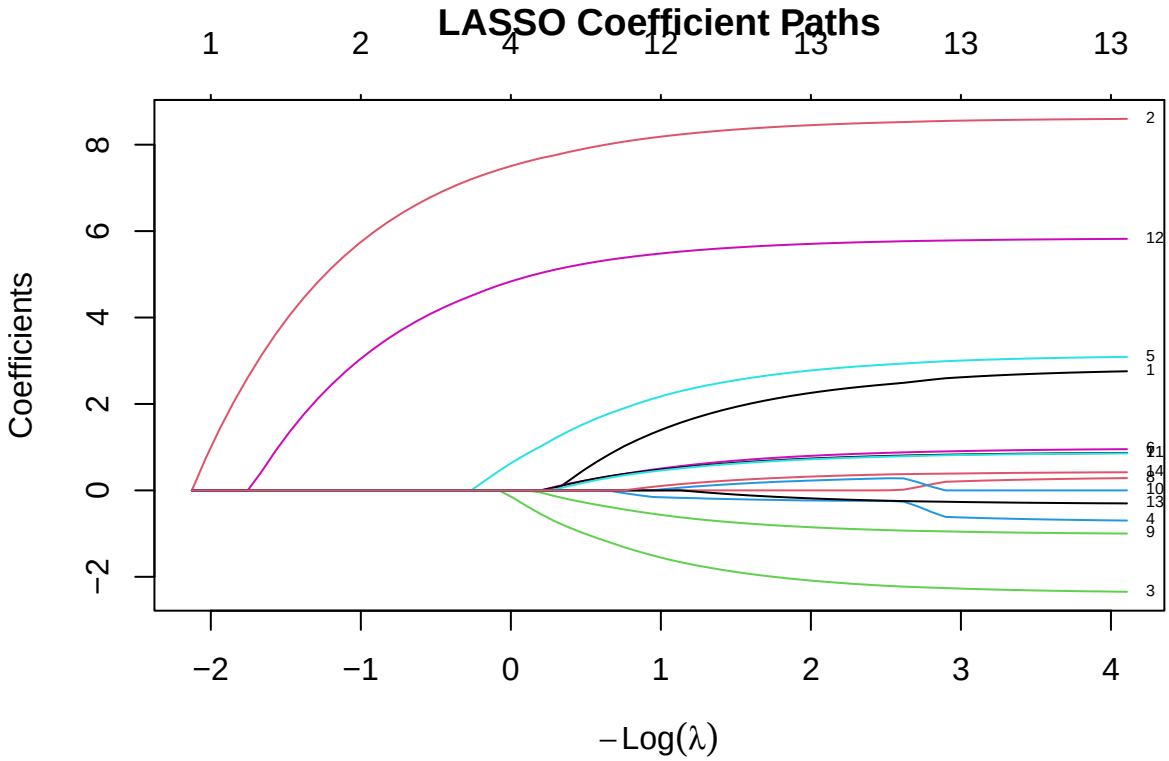
```
coef(lasso_fit, s = "lambda.min")
```

```
## 15 x 1 sparse Matrix of class "dgCMatrix"
##                         lambda.min
## (Intercept)           45.2619175
## DiabetesComplications  2.6340494
## GLP1                   8.5610646
## Metformin             -2.2810274
## HOMAB                 -0.6339552
## Sex                    3.0142631
## PYY                    0.9147268
## Age                    0.8356951
## BMI                    0.2198876
## HbA1c                 -0.9607997
## MatsudaIdx             .
## Gastrin                0.8241557
## Ghrelin                5.7910326
## Amylin                -0.2719751
## Glucagon               0.3952961
```

```
plot(lasso_fit)
title("LASSO Cross-validation Curve")
```




```
plot(glmnet(x, y, alpha=1), xvar="lambda", label=TRUE)
title("LASSO Coefficient Paths")
```



Compared to model4 of linear regression model, repid change occured in the estimated impact of vars that has colinear relation with others in normalized linear regression model.

Conclusion:

In this study, we investigated factors influencing gastric emptying (GE) in individuals with Type II Diabetes Mellitus using the DiGeHormone dataset. Our analysis combined extensive data exploration with multivariable regression modeling, including standardization, covariate adjustment, interaction assessment, and regularization techniques.

Data Exploration:

Initial data exploration showed minimal missing data (<1%), which were removed without affecting results. Most biomarkers exhibited approximately normal distributions, whereas some clinical variables were slightly skewed, highlighting the importance of checking model assumptions. Categorical data was found to affect levels of some of the continuous value, indicating its relevance as a covariate. Strong correlations were observed between BMI and body weight, HbA1c and fasting glucose, and other related measures, suggesting potential multicollinearity among predictors. Scatterplots indicated weak to moderate linear relationships between most biomarkers and GE, supporting the use of multivariable regression models.

Modeling Results:

Multivariable linear regression identified several key factors associated with GE. Sex and Metformin usage showed significant effects, with males and non-users having faster gastric emptying. Ghrelin and GLP1 were strongly positively associated with GE, while Gastrin and PYY demonstrated marginal associations. Covariate inclusion and standardization improved model stability and reduced collinearity effects, allowing retention of biologically relevant variables such as Matsuda Index and HOMA-B.

Interaction analyses revealed that sex modulates the effect of PYY on GE, with a trend-level interaction suggesting the positive association between PYY and GE is stronger in females. Furthermore, while DiabetesComplications alone was not a significant predictor, its interaction with GLP1 was significant, indicating a moderation effect where GLP1's positive impact on GE is attenuated in patients with complications.

Finally, regularization using LASSO confirmed the robustness of key predictors, highlighting the importance of Ghrelin, GLP1, Metformin, and Sex in explaining variability in GE while controlling for multicollinearity. LASSO coefficient paths and cross-validation curves provided additional insight into variable selection and model stability.

Overall, our results suggest that endogenous GI hormones, medication usage, and sex are important determinants of gastric emptying in T2DM. Interaction effects underscore the need to consider potential moderators in modeling physiological outcomes. Standardization and regularization techniques enhance model interpretability and reliability, particularly in the presence of collinear predictors.

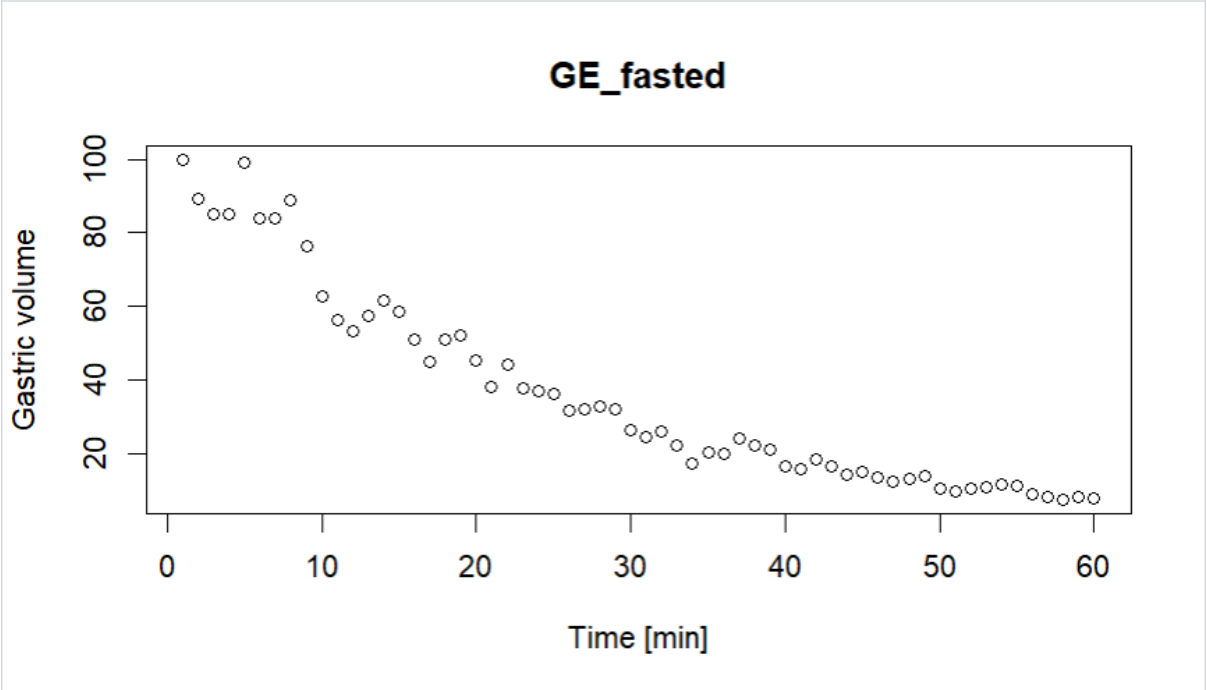
Task 2:

Q1. Model fasted gastric emptying (GE_fasted) to determine the rate and half-life of gastric emptying.

We start with looking at the data. We see that the median and mean deviate substantially for both GE_fasted and GE_fed suggesting skewness.

```
> summary(data_2)
       X                Time          GE_fasted          GE_fed       
 Min.   : 1.00   Min.   : 1.00   Min.   :  7.23   Min.   : -2.99  
 1st Qu.:15.75   1st Qu.:15.75   1st Qu.: 13.98   1st Qu.: 30.96  
 Median :30.50   Median :30.50   Median : 26.02   Median : 80.47  
 Mean   :30.50   Mean   :30.50   Mean   : 36.33   Mean   : 63.18  
 3rd Qu.:45.25   3rd Qu.:45.25   3rd Qu.: 52.37   3rd Qu.: 94.06  
 Max.   :60.00   Max.   :60.00   Max.   :100.00   Max.   :101.70  
```

Plotting the fasted data reveals a decay trend, consistent with an exponential process. We therefore fit an exponential model.



We assume the data to follow a decay curve, and we fit a model for it.

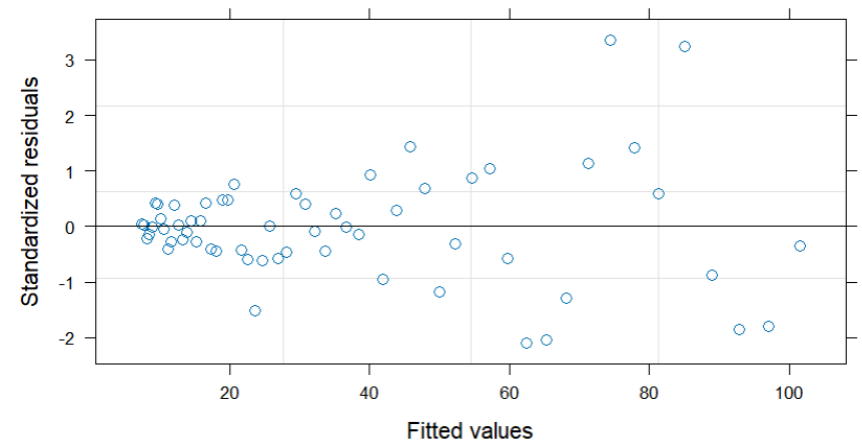
```r
#we try to fit an exponential decay model
fasted_fit <- nls(GE_fasted ~ GE0 * exp(-k * X),
                  data = data_2,
                  start = list(GE0 = max(data_2$GE_fasted), k = 0.01))
```

```
> summary(fasted_fit)

Formula: GE_fasted ~ GE0 * exp(-k * X)

Parameters:
     Estimate Std. Error t value Pr(>|t|)
GE0 1.061e+02  1.964e+00   54.01   <2e-16 ***
k   4.416e-02  1.188e-03   37.17   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.337 on 58 degrees of freedom

Number of iterations to convergence: 5
Achieved convergence tolerance: 2.263e-06
```

The model summary shows:

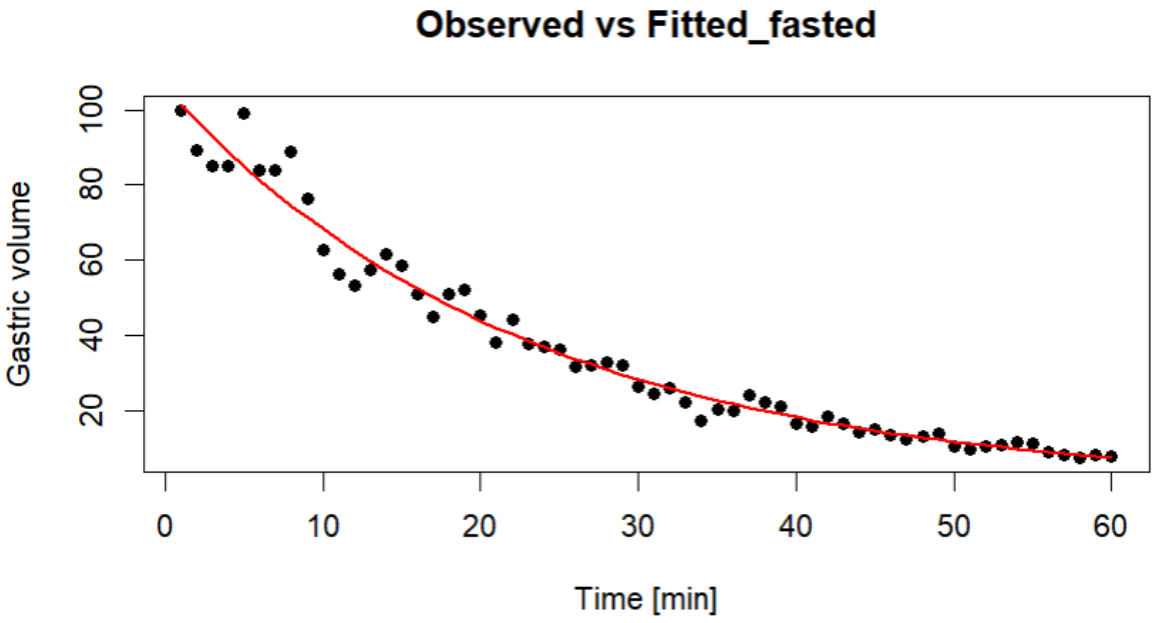
- Low standard errors
- High t-values
- Statistically significant p-values
- Low residuals

Together, these indicate a good model fit. A residuals vs. fitted plot confirms that residuals are small and evenly distributed.



Comparing predictions against observed data shows that the predicted curve (red line) closely follows the original data points (black dots).

```
time_seq <- seq(min(data_2$X), max(data_2$X), length.out = 60)
pred_fasted <- predict(fasted_fit, newdata_fasted = data.frame(X = time_seq))

plot(data_2$X, data_2$GE_fasted, pch = 16, main = "Observed vs Fitted_fasted",  xlab
lines(time_seq, pred_fasted, col = "red", lwd = 2)
```



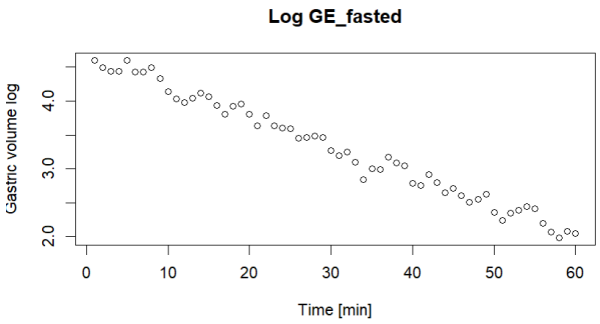
From this model we determine:

- **Rate constant (k):** 0.0044
- **Half-life ($t_{1/2}$):** 15.7

```
> coef_fasted <- coef(fasted_fit)
> t_half_fasted <- log(2)/coef_fasted["k"]
> print(t_half_fasted)
       k
15.69482
```

We could also use an alternative approach with log transforming the data.

```
log_GE_fasted <- log(data_2$GE_fasted)

#We see that we get a linear relationship
plot(data_2$X,log_GE_fasted, main = "Log GE_fasted", xlab = "Time [min]", ylab = "Ga
```



We can now plot the log data, we see that the data seem to follow a more linear trend. We can now fit a linear model to the data.

```
> lm_GE_fasted <- lm(formula = log_GE_fasted ~ X, data=data_2)
> summary(lm_GE_fasted)

Call:
lm(formula = log_GE_fasted ~ X, data = data_2)

Residuals:
      Min        1Q    Median        3Q       Max
-0.315603 -0.079253  0.001673  0.073441  0.185133

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.6576842  0.0266059   175.1   <2e-16 ***
X           -0.0442232  0.0007586   -58.3   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1018 on 58 degrees of freedom
Multiple R-squared:  0.9832,    Adjusted R-squared:  0.9829
F-statistic:  3399 on 1 and 58 DF,  p-value: < 2.2e-16
```

The residuals are normally distributed, standard errors are low, and the R^2 value is close to 1, confirming a strong fit. The estimated rate and half-life are nearly identical to the exponential fit.

```
> AIC(fasted_fit)
[1] 350.2964
> AIC(lm_GE_fasted)
[1] -99.97951
```

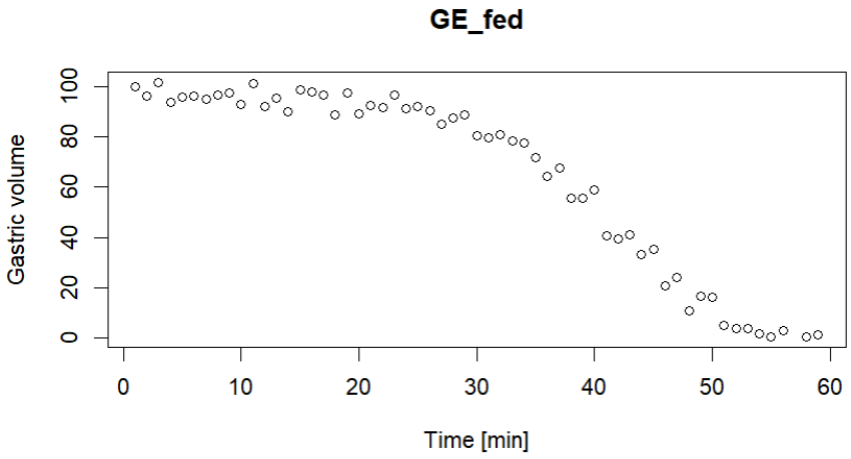
We can also compare the models with the AIC values for the models. Since lm_GE_fasted has a lower AIC value the model is a better fit.

Q2. Model the fed state gastric emptying profile.

When looking at the data set for the fed_state we see two negative values. Negative volume is not possible so we remove them from the data.

```
GE_fed_data <- data_2$GE_fed[data_2$GE_fed >= 0]
length(GE_fed_data)
X_data <- data_2$X[data_2$GE_fed > 0]
```

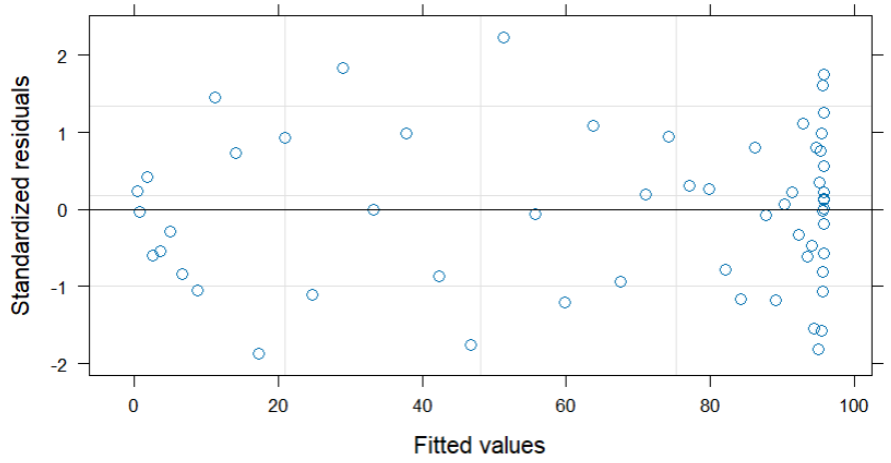
Plotting the fed-state data shows that the profile does not follow a simple exponential decay. Instead, there is an initial lag phase before emptying begins. To account for this, we fit a Weibull model, which incorporates a lag parameter.



```
> summary(fed_fit)

Formula: GE_fed_data ~ GE0 * exp(-(k * X_data)^b)

Parameters:
     Estimate Std. Error t value Pr(>|t|)
GE0 9.570e+01  7.172e-01  133.43   <2e-16 ***
k   2.296e-02  1.176e-04  195.27   <2e-16 ***
b   5.529e+00  2.159e-01   25.61   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.424 on 55 degrees of freedom

Number of iterations to convergence: 7
Achieved convergence tolerance: 1.058e-06
```

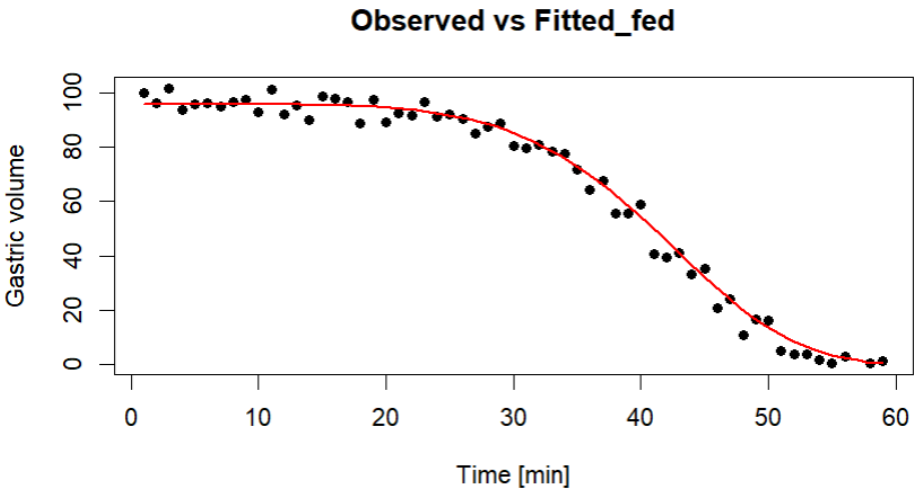
The Weibull model shows:

- Low standard errors
- Statistically significant p-values
- Low residuals




```r
# Predictions
time_seq_fed <- seq(min(X_data), max(X_data), length.out = 58)
pred_fed <- predict(fed_fit, newdata = data.frame(X = time_seq_fed))

# Plot
plot(X_data, GE_fed_data, pch = 16, main = "Observed vs Fitted_fed",
     xlab = "Time [min]", ylab = "Gastric volume")
lines(time_seq_fed, pred_fed, col = "red", lwd = 2)
```

Predicted vs. observed plots demonstrate that the Weibull model fits the data well across the time course.



Q3. *In larger studies we normally depend on secondary parameters (such as half-life) rather than the time-dynamic profiles.*
For the fed state gastric emptying profile, how informative is half-life for describing the time-dynamics?
Can you propose alternative secondary parameters?

In larger studies, secondary parameters (such as half-life) are often used instead of full time-dynamic profiles. However, half-life is not always informative for the fed state because:

- Different curve shapes can yield the same t_{50}.
- Subjects may share identical t_{50} values but differ in early retention, which has important clinical implications.
- Half-life assumes exponential decay and does not account for the lag phase.

For example:

- **$T_{1/2}$ (fasted):** 15.7
- **$T_{1/2}$ (fed):** 30.2

This comparison suggests that fed emptying is only about twice as slow, when in fact the lag phase makes it much less representative.

```
> coef_fed <- coef(fed_fit)
> t_half_fed <- log(2)/coef_fed["k"]
> print(t_half_fed)
       k
30.19421
```

Example of another parameter.
T-lag25- time until 25% of gastric volume is emptied

```
> threshold_fed <- 0.75 * coef_fed["GE0"]
> idx_fed <- which.min(abs(pred_fed - threshold_fed))
> T25_fed<- time_seq_fed[idx_fed]
> print(T25_fed)
[1] 35.59649
```

.

```
> threshold_fasted <- 0.75 * coef_fasted["GE0"]
> idx_fasted <- which.min(abs(pred_fasted- threshold_fasted))
> T25_fasted <- time_seq[idx_fasted]
> print(T25_fasted)
[1] 7
```

Here we get a much better description of the lag in the fed state and how much faster the fasted state is.

Example of another parameter:

*Time at Steepest decline (most negative derivate)

*Area under the curve (AUC)

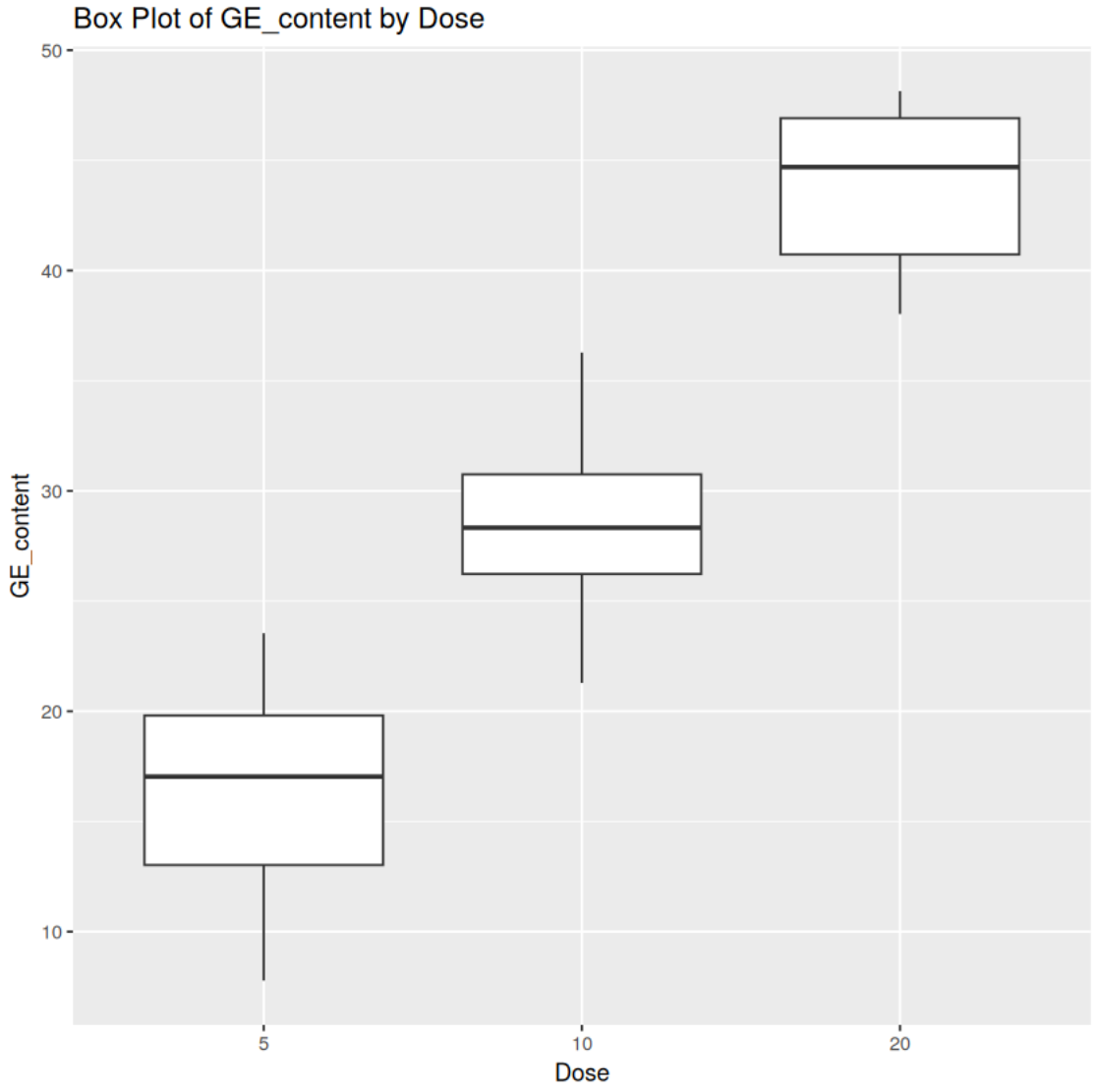
Conclusion

- **Fasted gastric emptying** is well described by an exponential (or log-linear) model with a half-life of ~15.7 minutes.
- **Fed gastric emptying** requires a Weibull model due to the initial lag phase; half-life (~30.2 minutes) is insufficient to describe its dynamics.
- **Half-life is not always informative** in the fed state. Alternative secondary parameters such as *T-lag25* and *time at steepest decline, AUC* provide a more accurate representation of gastric emptying dynamics.

Task_3_a

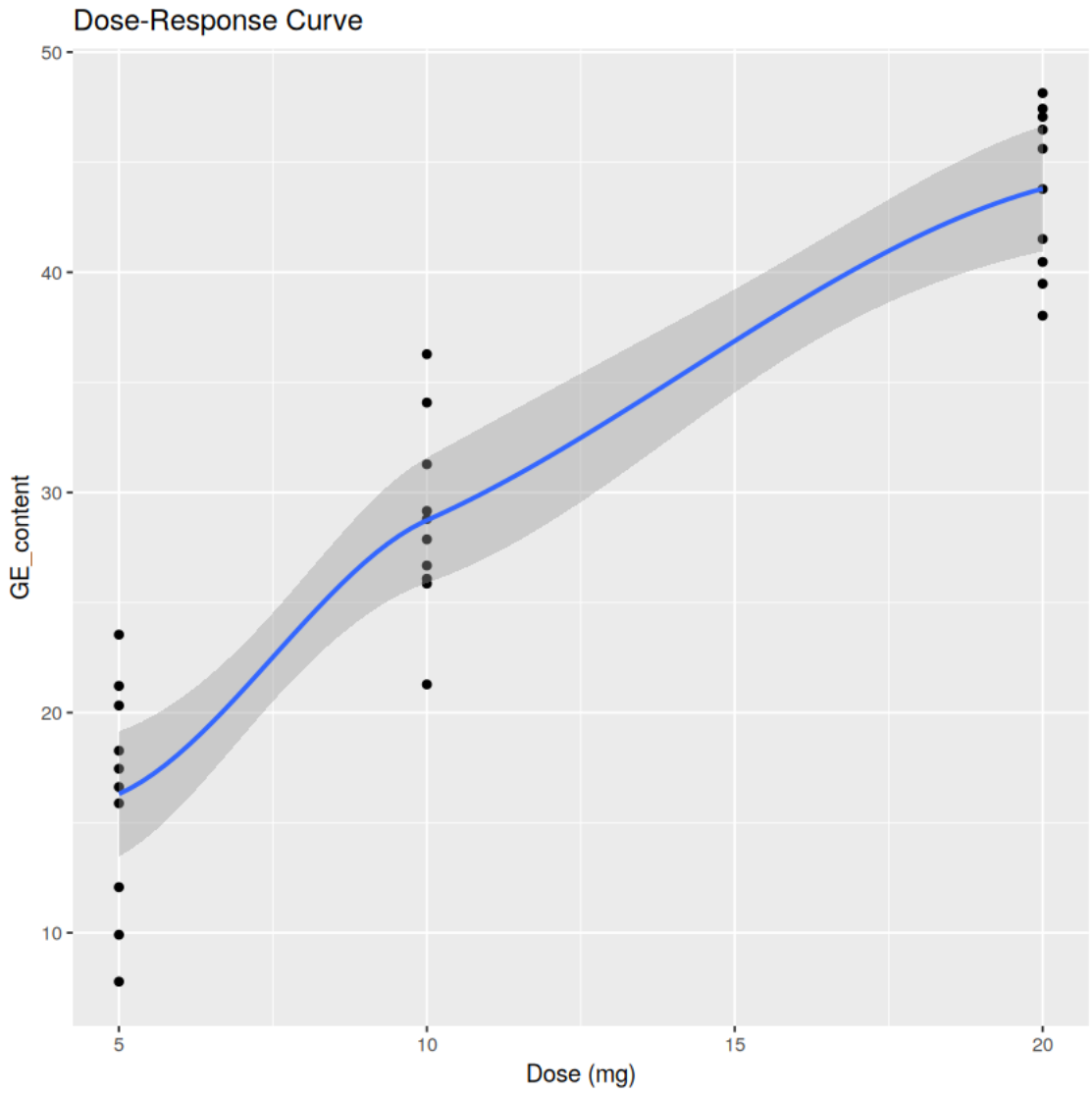
October 8, 2025

[107]:
```
ggplot(data_low, aes(x = factor(Dose), y = GE_content)) +
  geom_boxplot() +
  labs(title = "Box Plot of GE_content by Dose", x = "Dose", y = "GE_content")
```

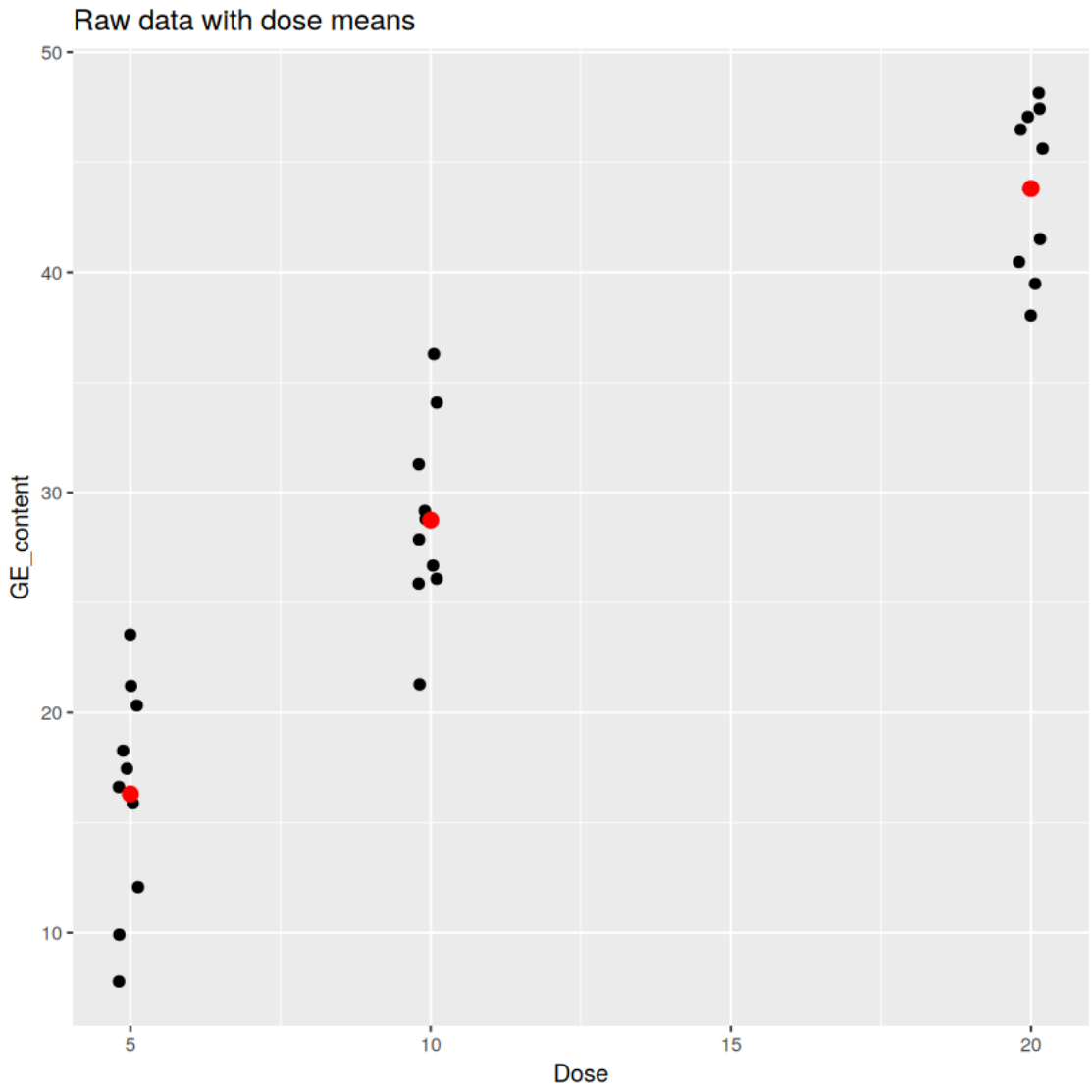


[108]:
```r
# Plot the data
ggplot(data_low, aes(x = Dose, y = GE_content)) +
  geom_point() +
  geom_smooth(method = "loess") +
  labs(title = "Dose-Response Curve", x = "Dose (mg)", y = "GE_content")
```

```
`geom_smooth()` using formula = 'y ~ x'
Warning message in simpleLoess(y, x, w, span, degree = degree, parametric =
parametric, :
"pseudoinverse used at 4.925"
Warning message in simpleLoess(y, x, w, span, degree = degree, parametric =
parametric, :
"neighborhood radius 15.075"
Warning message in simpleLoess(y, x, w, span, degree = degree, parametric =
parametric, :
"reciprocal condition number  1.2461e-16"
Warning message in simpleLoess(y, x, w, span, degree = degree, parametric =
parametric, :
"There are other near singularities as well. 227.26"
Warning message in predLoess(object$y, object$x, newx = if (is.null(newdata))
object$x else if (is.data.frame(newdata))
as.matrix(model.frame(delete.response(terms(object)), :
"pseudoinverse used at 4.925"
Warning message in predLoess(object$y, object$x, newx = if (is.null(newdata))
object$x else if (is.data.frame(newdata))
as.matrix(model.frame(delete.response(terms(object)), :
"neighborhood radius 15.075"
Warning message in predLoess(object$y, object$x, newx = if (is.null(newdata))
object$x else if (is.data.frame(newdata))
as.matrix(model.frame(delete.response(terms(object)), :
"reciprocal condition number  1.2461e-16"
Warning message in predLoess(object$y, object$x, newx = if (is.null(newdata))
object$x else if (is.data.frame(newdata))
as.matrix(model.frame(delete.response(terms(object)), :
"There are other near singularities as well. 227.26"
```



[112]:
```
ggplot(data_low, aes(Dose, GE_content)) +
  geom_jitter(width = 0.2, height = 0, size = 2) +
  stat_summary(fun = mean, geom = "point", colour = "red", size = 3) +
  labs(title = "Raw data with dose means")
```



[113]:
```
library(drc)

ml1 <- drm(GE_content ~ Dose, data = data_low,
fct = LL.3(names = c("Hill slope", "Min", "Max")))
```

[114]:
```
summary(ml1)
```

```
Model fitted: Log-logistic (ED50 as parameter) with lower limit at 0 (3 parms)

Parameter estimates:

                  Estimate Std. Error t-value  p-value
```

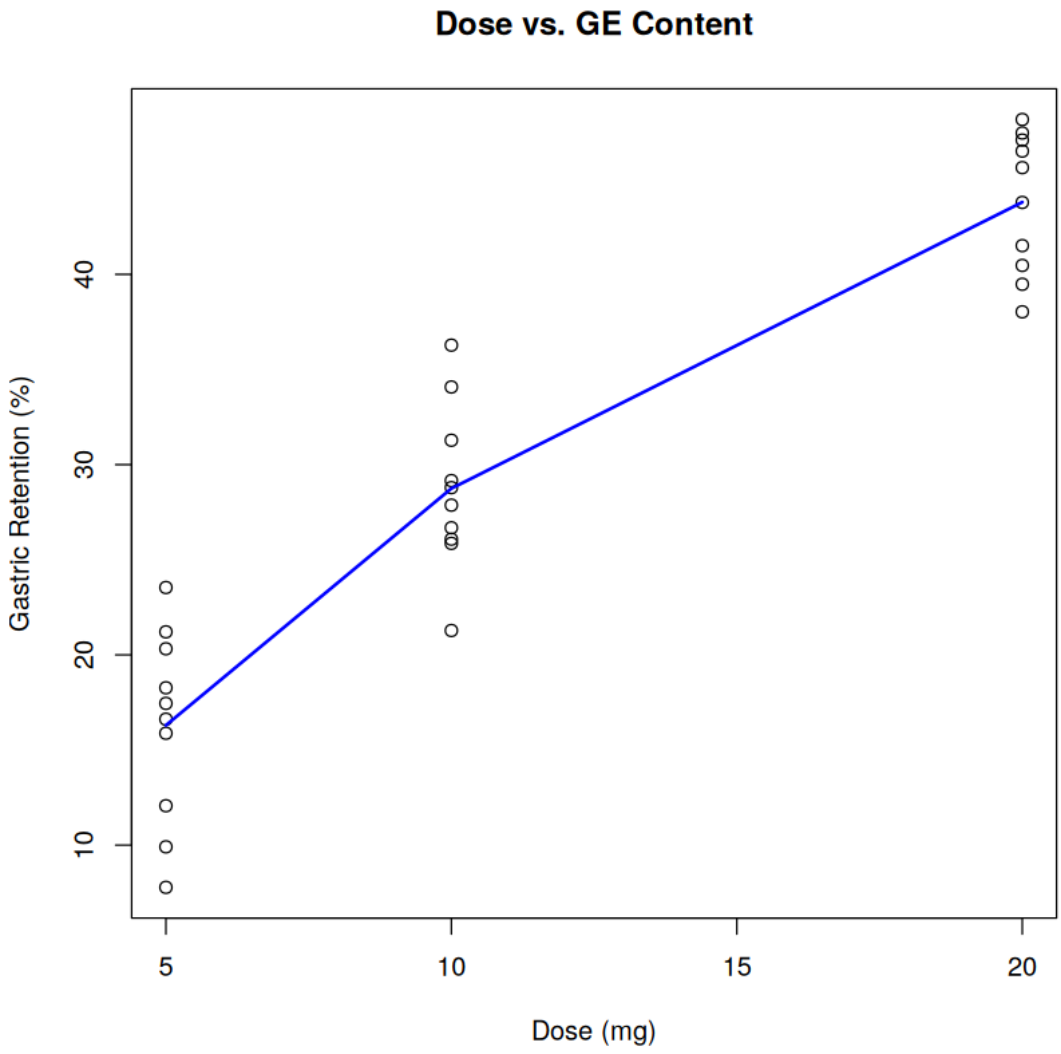
```
Hill slope:(Intercept) -1.15247    0.41204 -2.7970 0.009393 **
Min:(Intercept)        76.56995   37.85265  2.0228 0.053095 .
Max:(Intercept)        15.55276   13.87981  1.1205 0.272356
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error:

 4.38386 (27 degrees of freedom)
```

[115]:
```
summary(ml1)[[3]]
```

A matrix: 3 × 4 of type dbl

	Estimate	Std. Error	t-value	p-value
Hill slope:(Intercept)	-1.152473	0.4120441	-2.796964	0.009392691
Min:(Intercept)	76.569954	37.8526532	2.022842	0.053095391
Max:(Intercept)	15.552757	13.8798116	1.120531	0.272355939

[116]:
```
# Plot the data and the fitted polynomial regression line
plot(data_low$Dose, data_low$GE_content, main = "Dose vs. GE Content", xlab =␣
 ↪"Dose (mg)", ylab = "Gastric Retention (%)")
lines(sort(data_low$Dose), predict(ml1, newdata = data[order(data_low$Dose),␣
 ↪]), col = "blue", lwd = 2)
```



[117]:
```
ED(ml1, 50)
```

```
Estimated effective doses

       Estimate Std. Error
e:1:50   15.553     13.880
```

15.553 dose to produce a retention of 50% of gastric content at 30 minutes.

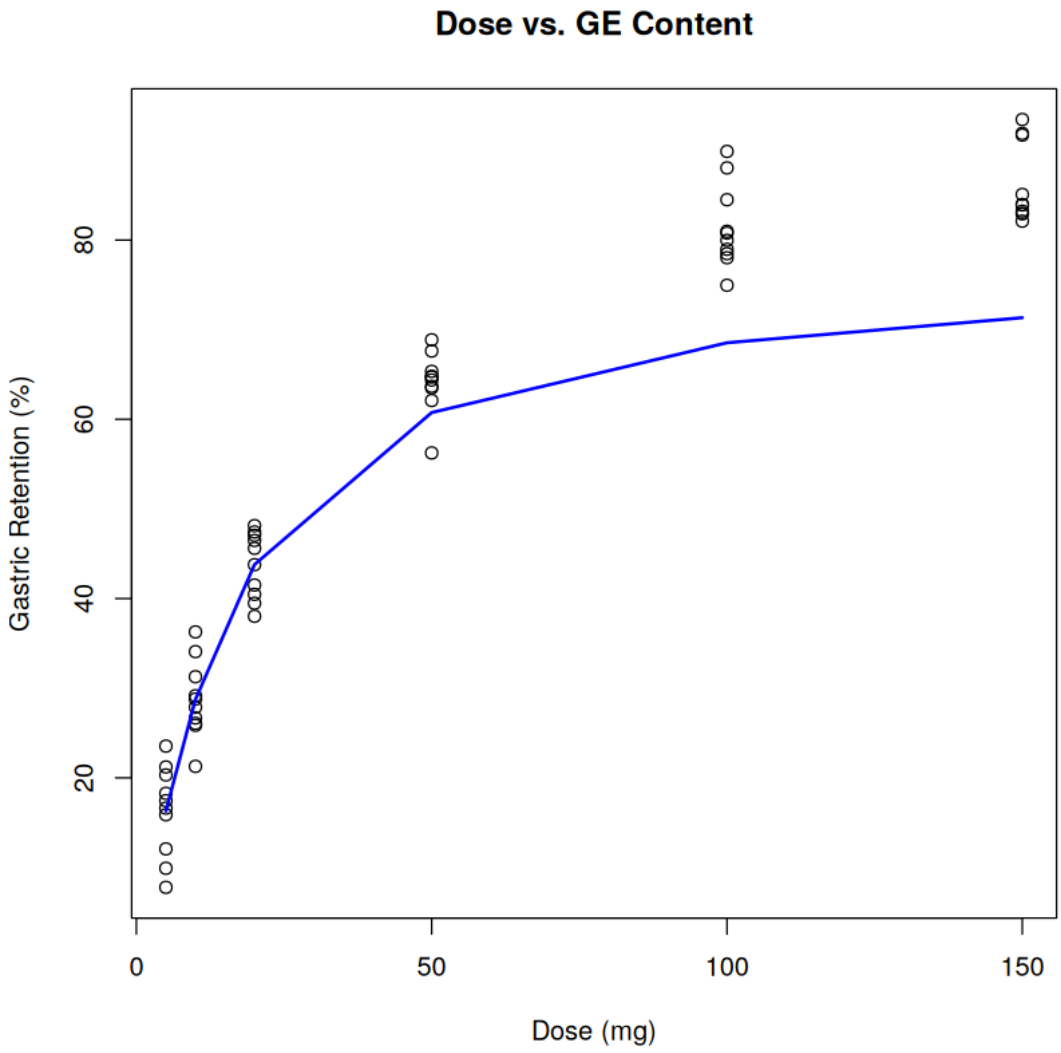
[118]:
```
ED(ml1, 80)
```

```
Estimated effective doses
```

```
      Estimate Std. Error
e:1:80   51.786     68.174
```

51.786 the dose that produces 80% gastric retention at 30 minutes.

[119]:
```
# Plot the data and the fitted LL.3 regression line
plot(data$Dose, data$GE_content, main = "Dose vs. GE Content", xlab = "Dose␣
 ↪(mg)", ylab = "Gastric Retention (%)")
lines(sort(data$Dose), predict(ml1, newdata = data[order(data$Dose), ]), col =␣
 ↪"blue", lwd = 2)
```

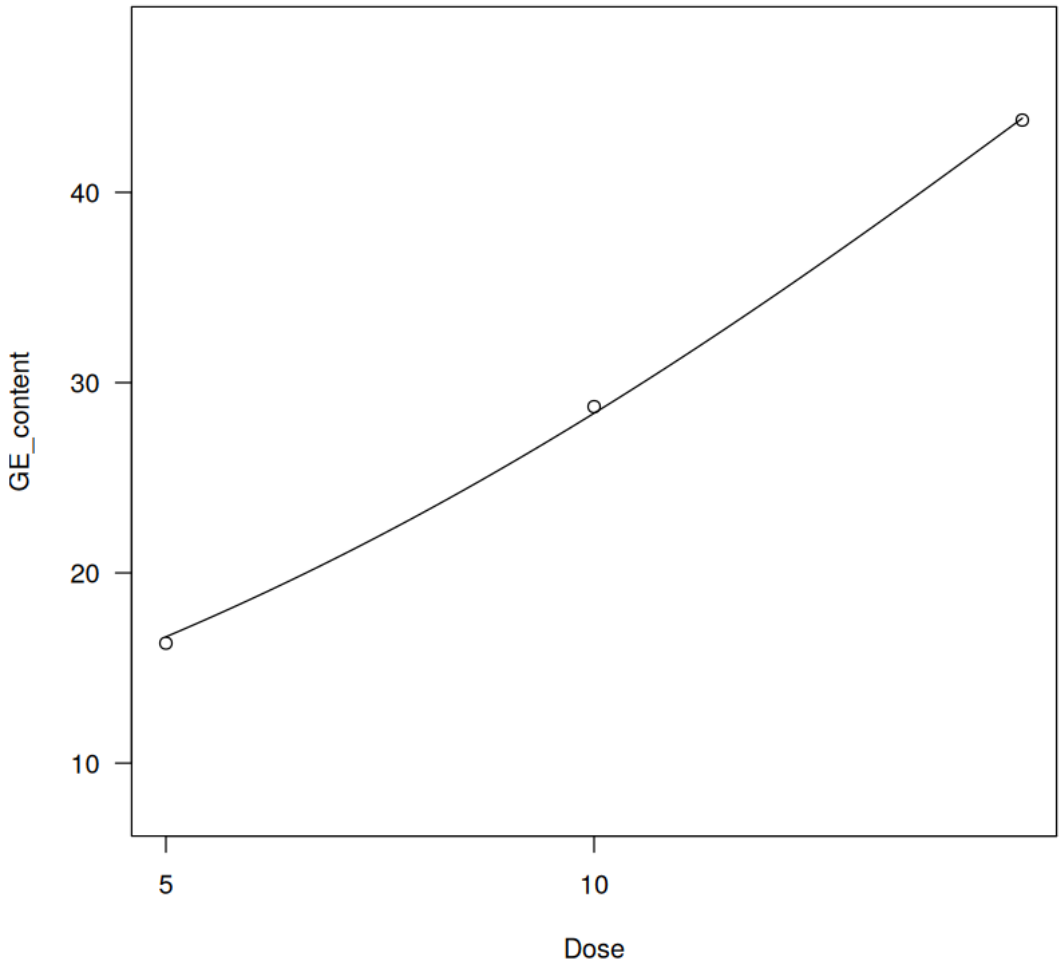


[120]:
```
k1 <- drm(GE_content ~ Dose, data = data_low,
fct = MM.2())
summary(k1)
plot(k1)
AIC(ml1, k1)
```

```
Model fitted: Michaelis-Menten (2 parms)

Parameter estimates:

              Estimate Std. Error t-value   p-value
d:(Intercept)  96.6920    15.3002  6.3197 7.777e-07 ***
e:(Intercept)  24.0561     6.1135  3.9349 0.0005002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error:

 4.314761 (28 degrees of freedom)
```

A data.frame: 2 × 2

	df	AIC
	<dbl>	<dbl>
ml1	4	178.6513
k1	3	176.7890



Log-logistic (ED50 as parameter) with lower limit at 0, used the fct for Dose-Response Model. Michaelis-Menten fct can be used as a replacement for fct as it improves it by 1%.

[121]:
```
# getMeanFunctions()
```

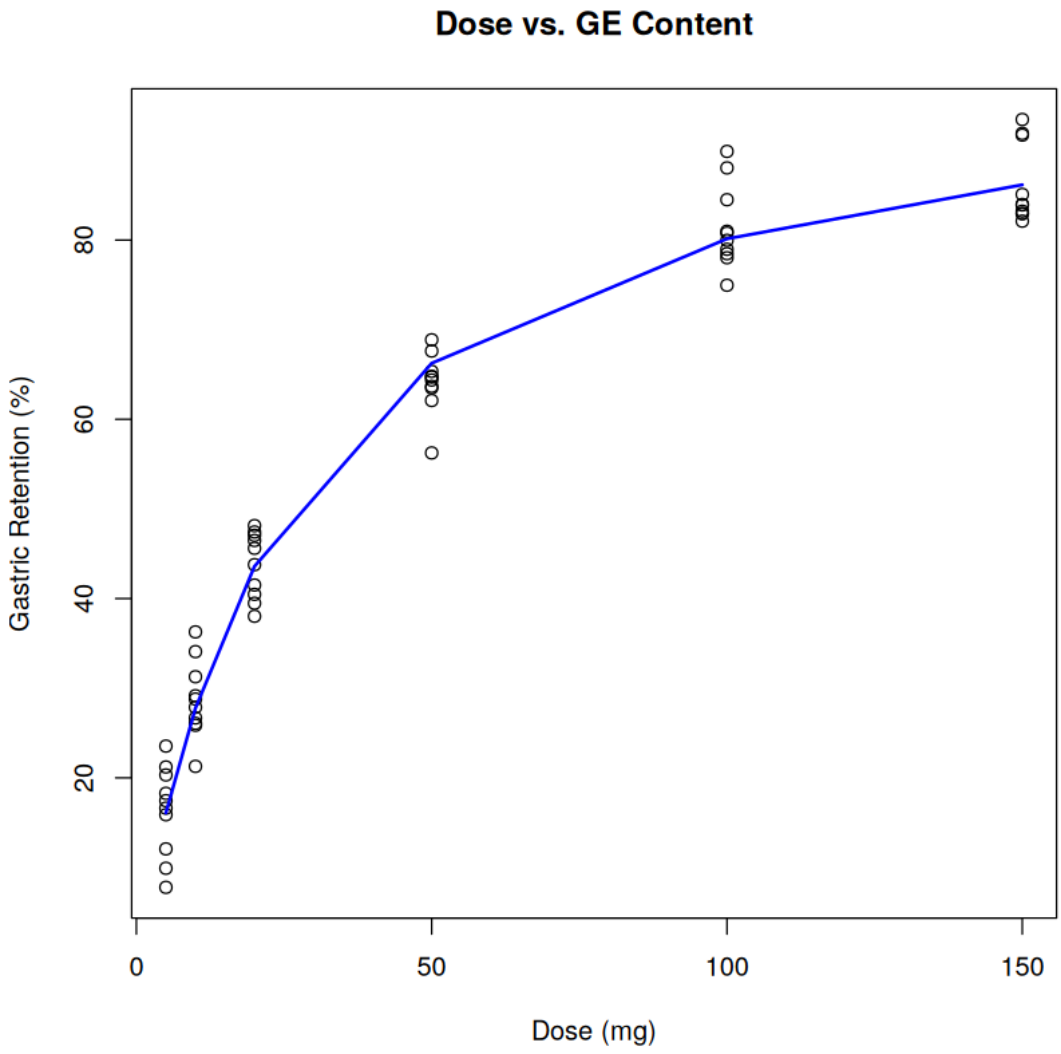
[122]:
```
# # Fit a polynomial regression model
# model_poly <- lm(GE_content ~ poly(Dose, 2), data = data)

# # Check the summary of the model
# summary(model_poly)
```

[123]:
```
ml2 <- drm(GE_content ~ Dose, data = data,
fct = MM.2())
```

[124]:
```
summary(ml2)
```

```
Model fitted: Michaelis-Menten (2 parms)

Parameter estimates:

              Estimate Std. Error t-value   p-value
d:(Intercept) 101.3913     1.8639  54.398 < 2.2e-16 ***
e:(Intercept)  26.5098     1.5221  17.417 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error:

 4.284886 (58 degrees of freedom)
```

[125]:
```
# Plot the data and the fitted polynomial regression line
plot(data$Dose, data$GE_content, main = "Dose vs. GE Content", xlab = "Dose␣
 ↪(mg)", ylab = "Gastric Retention (%)")
lines(sort(data$Dose), predict(ml2, newdata = data[order(data$Dose), ]), col =␣
 ↪"blue", lwd = 2)
```



[126]:
```
# Create a sequence of effects
effects <- seq(10, 90, by = 10)
# Estimate doses for each effect
doses <- numeric(length(effects))
for (i in 1:length(effects)) {
result <- ED(ml2, effects[i])
doses[i] <- result
}
# Create a data frame for plotting
df <- data.frame(Effect = effects, Dose = doses)
```

```
Estimated effective doses
```

```
       Estimate Std. Error
e:1:10  2.94554    0.16912

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses

       Estimate Std. Error
e:1:20  6.62746    0.38052

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses

       Estimate Std. Error
e:1:30 11.36136    0.65232

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses

       Estimate Std. Error
e:1:40  17.6732     1.0147

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses

       Estimate Std. Error
e:1:50  26.5098     1.5221

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses

       Estimate Std. Error
e:1:60  39.7648     2.2831

Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"


Estimated effective doses
```

```
       Estimate Std. Error
e:1:70  61.8563     3.5515
```

```
Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"
```

```
Estimated effective doses

       Estimate Std. Error
e:1:80 106.0394     6.0883
```

```
Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"
```

```
Estimated effective doses

       Estimate Std. Error
e:1:90  238.589     13.699
```

```
Warning message in doses[i] <- result:
"number of items to replace is not a multiple of replacement length"
```

[127]:
```r
# First, install plotly if you haven't already
# install.packages("plotly")

library(plotly)

# Assuming df is your data frame with Dose and Effect columns
# Create the interactive plot
p <- ggplot(df, aes(x = Dose, y = Effect)) +
  geom_line() +
  geom_point() +
  labs(x = "Dose", y = "Effect (%)")

# Convert to interactive plotly object
ggplotly(p, tooltip = c("x", "y"))
```

```
HTML widgets cannot be represented in plain text (need html)
```

Dose:106.0393 - Effect:80

Dosage level suggested for the follow-up experiment

Task4

zhexuan

2025-10-06

Task 4: The Follow-up Study of DiGeMon-123 in Rats

In this task, we model the data to investigate if the DiGeMon-123 has any effect on Lee index, and present and discuss our findings.

Load Data

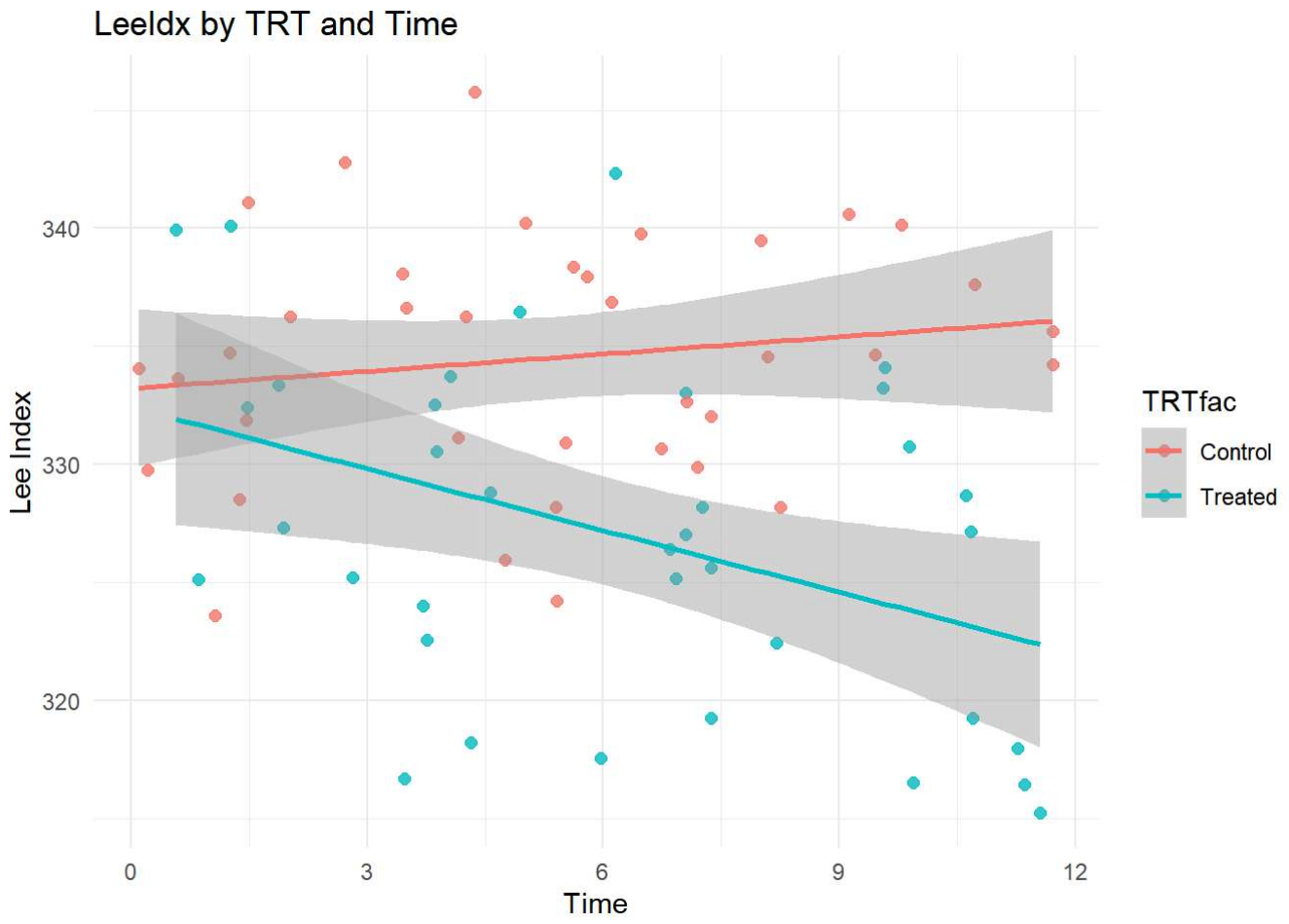
We first present the data4. It includes 73 observations, each representing one rat, with information on treatment group (TRT), treatment duration (Time, in weeks), and Lee Index (LeeIdx). The scatter plot and box plot of the raw data are plotted below.

```
df <- read.csv("Data_T4.csv", stringsAsFactors = FALSE)
names(df) <- make.names(names(df))
df <- df %>% rename(ID = X, LeeIdx = LeeIdx, TRT = TRT, Time = Time)
df$ID <- as.factor(df$ID)
df$TRT <- as.numeric(df$TRT)   # 0/1 numeric
df$TRTfac <- factor(df$TRT, levels = c(0,1), labels = c("Control","Treated"))


print(df %>% group_by(TRTfac) %>% summarise(n = n(), mean_Lee = mean(LeeIdx, na.rm=TRUE), sd_Le
e = sd(LeeIdx, na.rm=TRUE)))
```

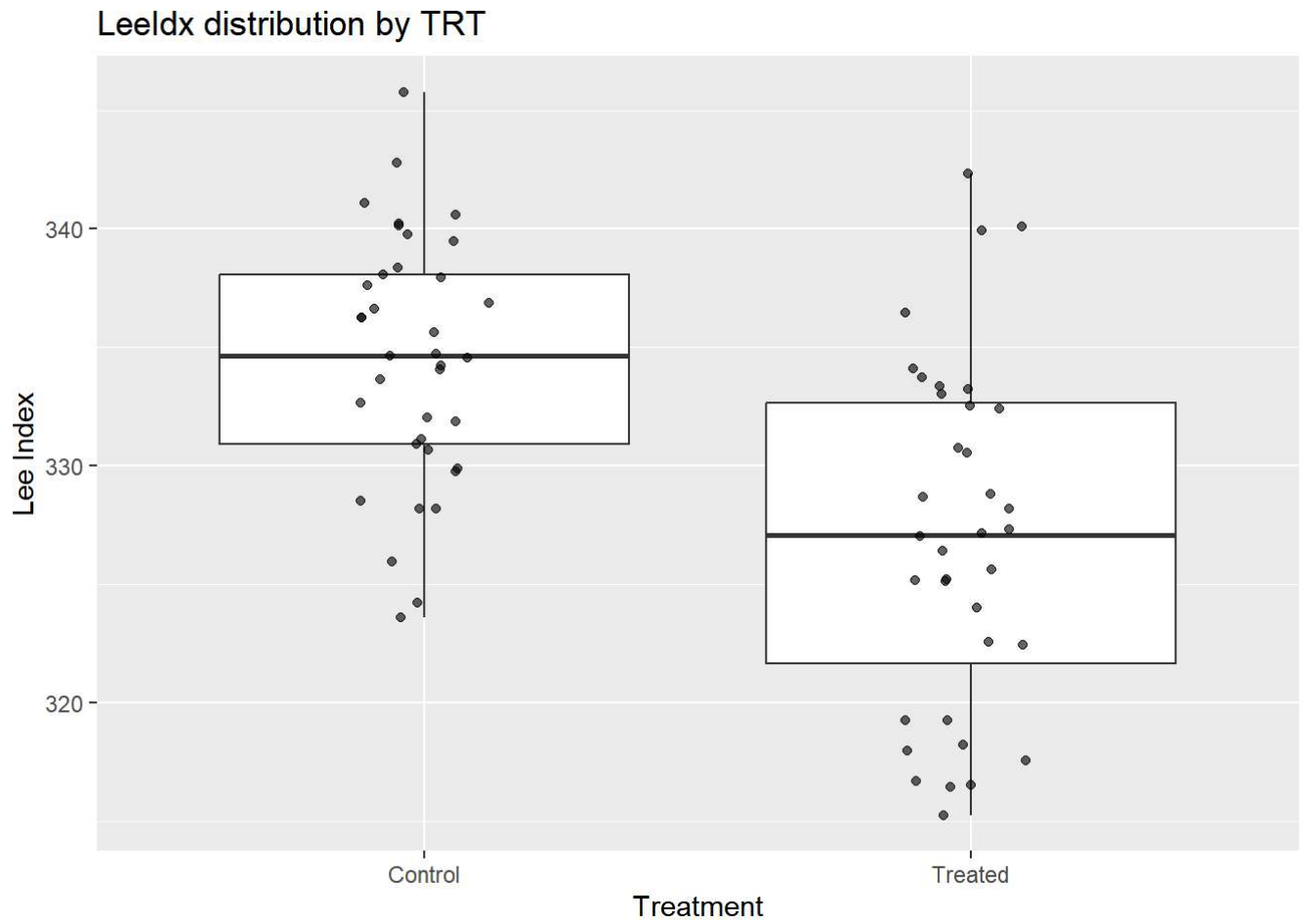
```
## # A tibble: 2 × 4
##   TRTfac      n mean_Lee sd_Lee
##   <fct>   <int>    <dbl>  <dbl>
## 1 Control    37     335.   5.20
## 2 Treated    36     327.   7.30
```

```
p1 <- ggplot(df, aes(x = Time, y = LeeIdx, color = TRTfac)) +
  geom_point(size = 2, alpha = 0.8) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(title = "LeeIdx by TRT and Time", x = "Time", y = "Lee Index") +
  theme_minimal()
print(p1)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# scatter plot & box plot
p2 <- ggplot(df, aes(x = TRTfac, y = LeeIdx)) +
  geom_boxplot() + geom_jitter(width = 0.12, alpha = 0.6) +
  labs(title = "LeeIdx distribution by TRT", x = "Treatment", y = "Lee Index")
print(p2)
```



Model selection & Comparison

Then we need to choose appropriate models to fit the data. Since each individual was only measured once, LeeIdx is a continuous outcome variable, and from the LeeIdx visualization it seems that there is no significant non-linear change over time, a linear regression model is appropriate here. To examine the effects of treatment and time on the Lee Index, two linear models were first fitted. Model 1 included only the main effects of treatment group (TRTfac) and time, while Model 2 included an additional interaction term between treatment and time. The purpose of this comparison was to test whether the rate of change in Lee Index over time differed between the treated and control groups.

Model 1 imposes the assumption that “time has the same effect on both groups”. The treated group had on average a 7.19-unit lower Lee Index than the control group ($p < 0.001$). Time showed a small non-significant negative trend with p-value larger than 0.05, indicating that if groups are not distinguished, the overall change over time is not significant.

From the results of model 2, neither the main effect of treatment nor that of time was statistically significant on their own (both $p > 0.05$), suggesting that at the baseline level, neither factor alone could explain much variation in the Lee Index. However, the interaction term (TRTfac:Time) was significant ($p < 0.05$). This indicates that the effect of time on the Lee Index depends on the treatment group — in other words, the slopes of the two lines differ. Specifically, the treated group shows a steeper negative slope, implying that their Lee Index decreases faster over time compared to the control group. It decreased by approximately 1.11 per unit time.

To formally compare the two models, an ANOVA was conducted between Model 1 and Model 2. The result ($F = 6.65$, $p = 0.012$) shows that including the interaction term significantly improves model fit, confirming that the interaction effect is meaningful and should be retained. Thus, Model 2 was used for all subsequent analyses.

```
mod1 <- lm(LeeIdx ~ TRTfac + Time, data = df)
summary(mod1)
```

```
##
## Call:
## lm(formula = LeeIdx ~ TRTfac + Time, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.3086  -3.9353   0.6504   4.8526  15.2958
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   336.2384     1.5792 212.919  < 2e-16 ***
## TRTfacTreated  -7.1945     1.4812  -4.857 7.01e-06 ***
## Time           -0.3238     0.2241  -1.445    0.153
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.274 on 70 degrees of freedom
## Multiple R-squared:  0.2854, Adjusted R-squared:  0.265
## F-statistic: 13.98 on 2 and 70 DF,  p-value: 7.803e-06
```

```
# with interaction (whether the therapeutic effect changes over time)
mod2 <- lm(LeeIdx ~ TRTfac * Time, data = df)
summary(mod2)
```

```
##
## Call:
## lm(formula = LeeIdx ~ TRTfac * Time, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.6969  -4.6524   0.3027   4.0205  15.2771
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        333.2155     1.9186 173.676   <2e-16 ***
## TRTfacTreated       -0.7944     2.8613  -0.278    0.782
## Time                 0.2428     0.3078   0.789    0.433
## TRTfacTreated:Time  -1.1125     0.4313  -2.579    0.012 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.035 on 69 degrees of freedom
## Multiple R-squared:  0.3482, Adjusted R-squared:  0.3199
## F-statistic: 12.29 on 3 and 69 DF,  p-value: 1.563e-06
```

```
# Compare the two models (whether interaction is needed)
anova_mods <- anova(mod1, mod2)
print(anova_mods)
```

```
## Analysis of Variance Table
##
## Model 1: LeeIdx ~ TRTfac + Time
## Model 2: LeeIdx ~ TRTfac * Time
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     70 2755.5
## 2     69 2513.2  1    242.31 6.6528 0.01203 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Heteroscedasticity SE

Because the presence of heteroscedasticity (unequal variance of residuals) could affect the reliability of standard errors, heteroskedasticity-robust standard errors were computed for Model 2. Using robust SE helps ensure that statistical inference remains valid even if the residual variance is not constant across observations. The significance pattern remained consistent with the original model: the interaction term remained significant, while the main effects did not. This consistency strengthens confidence in the robustness of the interaction effect.

```
#  (heteroskedasticity-robust SE)
cov_hc <- vcovHC(mod2, type = "HC3")
robust_t <- coeftest(mod2, vcov. = cov_hc)
print(robust_t)
```

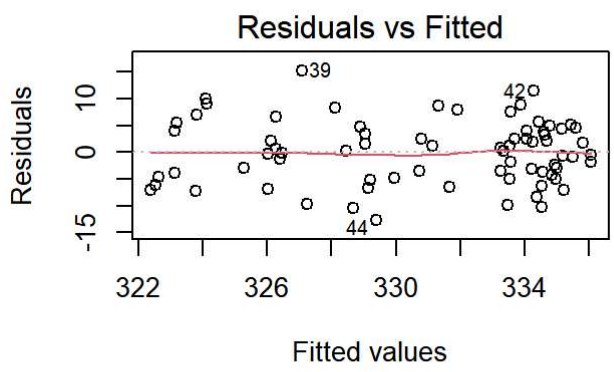
```
##
## t test of coefficients:
##
##                     Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)        333.21546    1.64514 202.5460 < 2.2e-16 ***
## TRTfacTreated       -0.79442    2.95471  -0.2689  0.788835
## Time                 0.24281    0.21769   1.1154  0.268557
## TRTfacTreated:Time  -1.11246    0.40621  -2.7387  0.007843 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

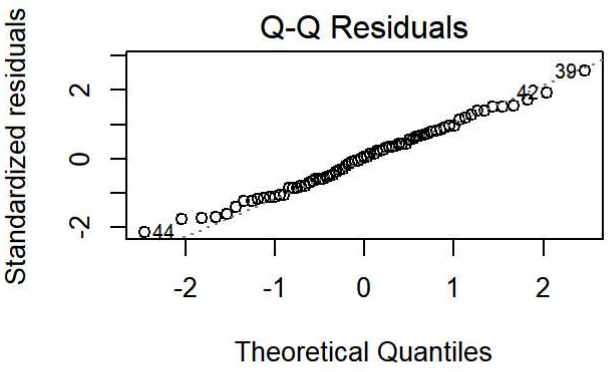
Analysis of Model 2

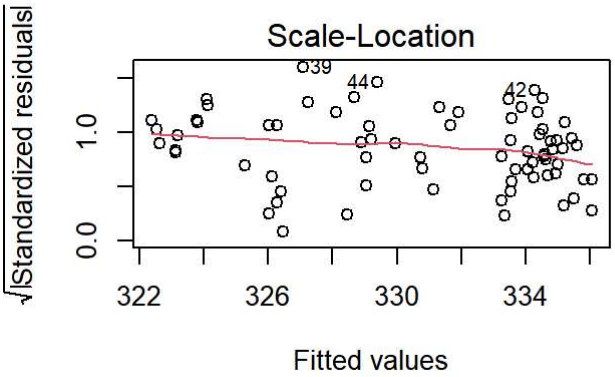
Model diagnostics were then performed to examine assumptions of linear regression. Four plots were generated: residuals vs fitted values, Q-Q plot, scale-location plot, and residuals vs leverage. The residual-fitted plot showed no clear pattern, suggesting approximate linearity and homoscedasticity. The Q-Q plot indicated that most residuals followed a normal distribution, with only minor deviations at the tails. The scale-location plot confirmed relatively constant variance, and the leverage plot revealed no extreme influential points. Overall, these diagnostics supported the adequacy of Model 2.

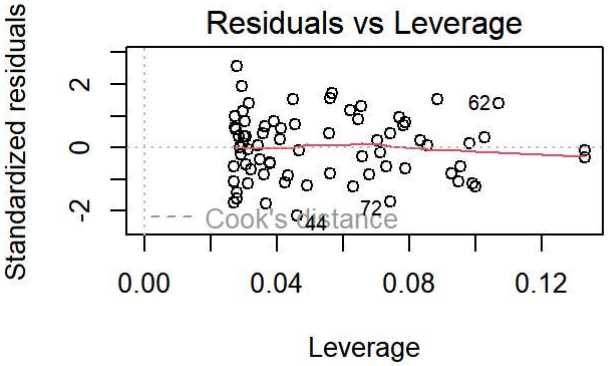
To further confirm residual normality, the Shapiro–Wilk test was conducted. The result ($p > 0.05$) indicated no significant departure from normality. The Breusch–Pagan test (BP = 3.97, df = 3, $p = 0.26$) was then used to assess heteroscedasticity. Since the p-value was greater than 0.05, the null hypothesis of homoscedasticity could not be rejected, suggesting that the variance of residuals was relatively constant across fitted values.

```
# includes Residuals vs Fitted, QQ, Scale-Location, Cook's
par(mfrow=c(2,2))
plot(mod2)
```










```
par(mfrow=c(1,1))

# normality test (Shapiro-Wilk on residuals)
sh_res <- shapiro.test(residuals(mod2))
print(sh_res)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(mod2)
## W = 0.99151, p-value = 0.9098
```

```
# test for heteroscedasticity (Breusch-Pagan)
bp <- bptest(mod2)
print(bp)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mod2
## BP = 3.9726, df = 3, p-value = 0.2644
```

To interpret the treatment effects more clearly, predicted means of the Lee Index were calculated from Model 2 for each combination of treatment and time, followed by pairwise comparisons between groups. The results revealed that, although baseline differences between groups were small, the treated group demonstrated a

larger decline in the predicted Lee Index over time compared to the control group. This finding quantitatively supports the earlier observation from the significant interaction term.

```
# choose times of practical interest:
times_to_eval <- c(min(df$Time, na.rm=TRUE),
                   median(df$Time, na.rm=TRUE),
                   max(df$Time, na.rm=TRUE))
times_to_eval
```

```
## [1]  0.1063677  5.6242958 11.7129661
```

```
# get emmeans at those times and pairwise contrasts
library(emmeans)
emm_at_times <- emmeans(mod2, ~ TRTfac | Time, at = list(Time = times_to_eval))
emm_at_times
```

```
## Time =  0.106:
##  TRTfac  emmean    SE df lower.CL upper.CL
##  Control    333 1.890 69      329      337
##  Treated    332 2.090 69      328      337
##
## Time =  5.624:
##  TRTfac  emmean    SE df lower.CL upper.CL
##  Control    335 0.996 69      333      337
##  Treated    328 1.020 69      325      330
##
## Time = 11.713:
##  TRTfac  emmean    SE df lower.CL upper.CL
##  Control    336 2.200 69      332      340
##  Treated    322 1.950 69      318      326
##
## Confidence level used: 0.95
```

```
contrast_at_times <- contrast(emm_at_times, method = "pairwise")   # default: treated - control
summary(contrast_at_times, infer = c(TRUE, TRUE))
```

```
## Time =  0.106:
##  contrast          estimate   SE df lower.CL upper.CL t.ratio p.value
##  Control - Treated    0.913 2.82 69    -4.72     6.54   0.323  0.7473
##
## Time =  5.624:
##  contrast          estimate   SE df lower.CL upper.CL t.ratio p.value
##  Control - Treated    7.051 1.43 69     4.21     9.90   4.945  <.0001
##
## Time = 11.713:
##  contrast          estimate   SE df lower.CL upper.CL t.ratio p.value
##  Control - Treated   13.825 2.94 69     7.96    19.69   4.704  <.0001
##
## Confidence level used: 0.95
```

Non-parametric Methods

To validate this conclusion using distribution-free methods, two non-parametric analyses were also performed. Hedges's g was computed to estimate the standardized mean difference between groups. g = 1.169 indicates that there is large treatment effect in reducing Lee Index. In addition, the Wilcoxon rank-sum test was conducted to test whether the median Lee Index differed significantly between treatment groups. And the p-value is very low here, indicating that there was a significant difference in the distribution of LeeIdx between the control group and the treatment group.

Robust Regression Model

Finally, a robust regression model was fitted to further ensure that the results were not driven by outliers or influential data points. Unlike ordinary least squares (OLS), robust regression assigns less weight to extreme values, making parameter estimates more stable. The robust model yielded results similar to Model 2: the interaction term (TRTfac:Time) remained significant, whereas main effects were not. This consistency across models strengthens the reliability of the conclusion that the treatment modifies the rate of change in Lee Index over time.

```
# Cohen's d for TRT groups, unadjusted
cohen_d <- cohen.d(df$LeeIdx ~ df$TRTfac, hedges.correction = TRUE)
print(cohen_d)
```

```
##
## Hedges's g
##
## g estimate: 1.169161 (large)
## 95 percent confidence interval:
##     lower     upper
## 0.6694236 1.6688978
```

```
# Non-parametric comparison(Comparison of the median of the two groups) was used as a robustnes
s test
wilcox_res <- wilcox.test(LeeIdx ~ TRTfac, data = df)
print(wilcox_res)
```

```
##
##  Wilcoxon rank sum exact test
##
## data:  LeeIdx by TRTfac
## W = 1059, p-value = 6.883e-06
## alternative hypothesis: true location shift is not equal to 0
```

```
# robust regression for outliers
rob_mod <- lmrob(LeeIdx ~ TRTfac * Time, data = df)
summary(rob_mod)
```

```
##
## Call:
## lmrob(formula = LeeIdx ~ TRTfac * Time, data = df)
##  \--> method = "MM"
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.6874  -4.2985   0.2877   4.2961  15.4049
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        333.2297     1.5749 211.584  < 2e-16 ***
## TRTfacTreated       -0.6647     3.0411  -0.219  0.82764
## Time                 0.2440     0.2020   1.208  0.23128
## TRTfacTreated:Time  -1.1578     0.4257  -2.720  0.00826 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Robust residual standard error: 6.217
## Multiple R-squared:  0.347,   Adjusted R-squared:  0.3186
## Convergence in 9 IRWLS iterations
##
## Robustness weights:
##  6 weights are ~= 1. The remaining 67 ones are summarized as
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5188  0.8871  0.9469  0.9161  0.9844  0.9985
## Algorithmic parameters:
##        tuning.chi                bb        tuning.psi        refine.tol
##         1.548e+00         5.000e-01         4.685e+00         1.000e-07
##           rel.tol         scale.tol         solve.tol          zero.tol
##         1.000e-07         1.000e-10         1.000e-07         1.000e-10
##       eps.outlier             eps.x warn.limit.reject warn.limit.meanrw
##         1.370e-03         2.131e-11         5.000e-01         5.000e-01
##      nResample          max.it       best.r.s       k.fast.s           k.max
##            500              50              2              1            200
##    maxit.scale      trace.lev            mts     compute.rd fast.s.large.n
##            200              0           1000              0           2000
##                   psi           subsampling                   cov
##             "bisquare"         "nonsingular"         ".vcov.avar1"
## compute.outlier.stats
##                  "SM"
## seed : int(0)
```

Conclusion

In summary, the linear model analysis, diagnostic checks, non-parametric tests, and robust regression all lead to a consistent conclusion. The treatment group exhibited a faster decline in Lee Index over time compared to the control group. And the robustness of this finding across multiple analytical approaches provides strong support for the validity of the result.