# Study on Heart disease

2025-03-22

## Description of Dataset

```r
library(ggplot2)
library(corrplot)
library(ggpubr)
library(knitr)
library(dplyr)
dat<-read.csv('heart.csv')
```

The Heart Failure Prediction Dataset is available on Kaggle. It contains 11 clinical features collected from patients, which can be used to predict potential heart disease events.

Here is overview of target variable and 11 clinical features.

| Name | Type | Meaning |
|---|---|---|
| HeartDisease | Binary Response Variable | output class [1: Heart Disease, 0: Normal] |
| Age | Numerical Predictor | age of the patient (in years) |
| Sex | Categorical Predictor | sex of the patient [M: Male, F: Female] |
| ChestPainType | Categorical Predictor | type of chest pain [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic] |
| RestingBP | Numerical Predictor | resting blood pressure (in mm Hg) |
| Cholesterol | Numerical Predictor | serum cholesterol (in mm/dl) |
| FastingBS | Categorical Predictor | fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise] |
| RestingECG | Categorical Predictor | resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality, LVH: Left Ventricular Hypertrophy (per Estes' criteria)] |
| MaxHR | Numerical Predictor | maximum heart rate achieved [between 60 and 202] |
| ExerciseAngina | Categorical Predictor | exercise-induced angina [Y: Yes, N: No] |
| Oldpeak | Numerical Predictor | ST depression induced by exercise relative to rest |
| ST_Slope | Categorical Predictor | the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping] |

## Data Quality Assessment

```r
kable(sapply(dat, function(x) sum(is.na(x))),caption = "Variable Missingnes Check")
```

Table 2: Variable Missingnes Check

| | x |
|---|---|
| Age | 0 |
| Sex | 0 |
| ChestPainType | 0 |

| | x |
|---|---|
| RestingBP | 0 |
| Cholesterol | 0 |
| FastingBS | 0 |
| RestingECG | 0 |
| MaxHR | 0 |
| ExerciseAngina | 0 |
| Oldpeak | 0 |
| ST_Slope | 0 |
| HeartDisease | 0 |

```r
kable(table(dat$HeartDisease), caption = "Heart Disease Status Frequency")
```

Table 3: Heart Disease Status Frequency

| Var1 | Freq |
|---|---|
| 0 | 410 |
| 1 | 508 |

```r
dup<-sum(duplicated(dat))
cat("Duplicated rows:",dup)
```

```
## Duplicated rows: 0
```

The dataset contains 918 observations without any missing values or duplicate rows. The response variable is roughly balanced.

# Objectives

Our analysis aims to answer the following questions:

1. What factors in the dataset have significant impacts on probability of getting heart disease?
2. How and to what extent do those factors influence the probability of getting heart disease?

In addition, we will develop a statistical model to predict the probability of getting heart disease.

# Exploratory Data Analysis

## Numerical Summaries

```r
# Summary of numeric variables
kable(summary(dat[, c("Age", "RestingBP","Cholesterol", "MaxHR", "Oldpeak")]))
```

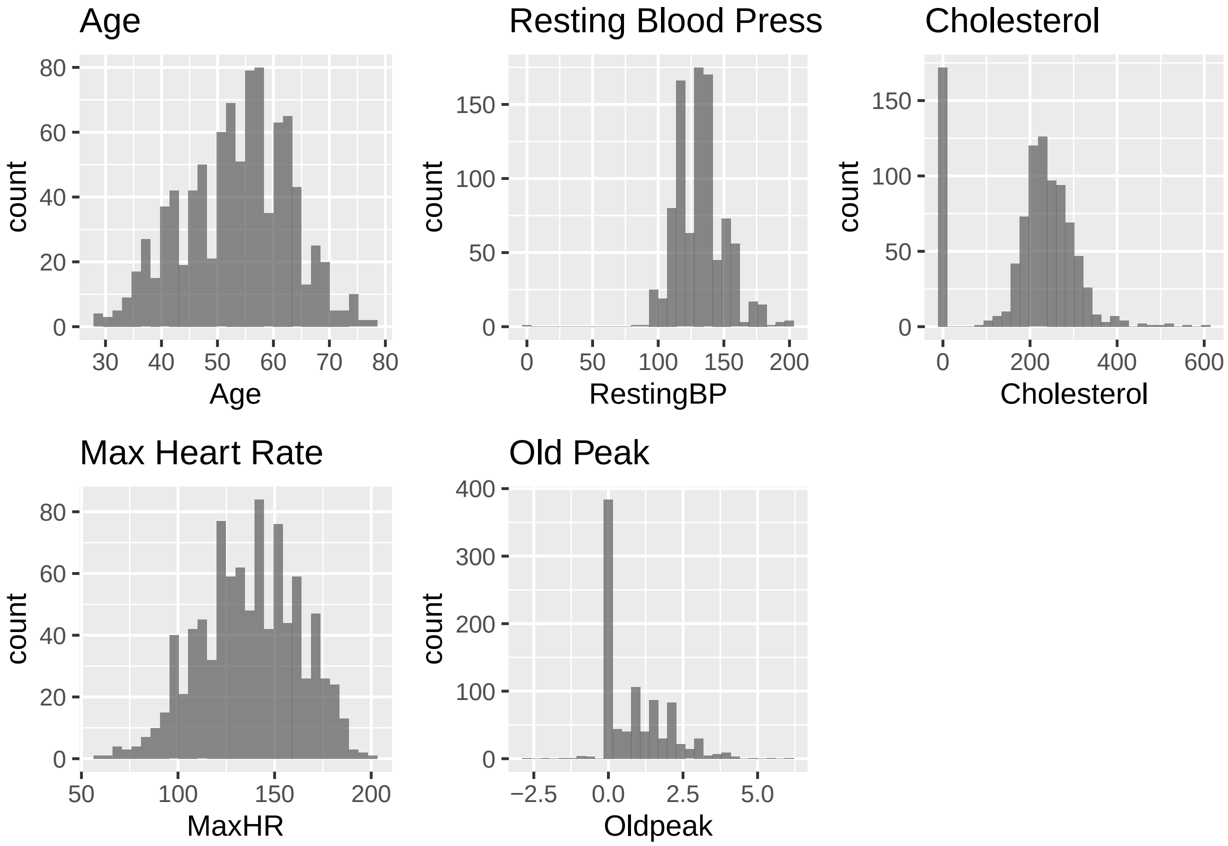
| | Age | RestingBP | Cholesterol | MaxHR | Oldpeak |
|---|---|---|---|---|---|
| | Min. :28.00 | Min. : 0.0 | Min. : 0.0 | Min. : 60.0 | Min. :-2.6000 |
| | 1st Qu.:47.00 | 1st Qu.:120.0 | 1st Qu.:173.2 | 1st Qu.:120.0 | 1st Qu.: 0.0000 |
| | Median :54.00 | Median :130.0 | Median :223.0 | Median :138.0 | Median : 0.6000 |
| | Mean :53.51 | Mean :132.4 | Mean :198.8 | Mean :136.8 | Mean : 0.8874 |
| | 3rd Qu.:60.00 | 3rd Qu.:140.0 | 3rd Qu.:267.0 | 3rd Qu.:156.0 | 3rd Qu.: 1.5000 |
| | Max. :77.00 | Max. :200.0 | Max. :603.0 | Max. :202.0 | Max. : 6.2000 |

| Name | Max | Min | Mean | Median |
|---|---|---|---|---|
| Age | 77 | 28 | 53.5108932 | 54 |
| RestingBP | 200 | 0 | 132.3965142 | 130 |
| Cholesterol | 603 | 0 | 198.7995643 | 223 |
| MaxHR | 202 | 60 | 136.8093682 | 138 |
| Oldpeak | 6.2 | -2.6 | 0.8873638 | 0.6 |

Oldpeak has negative value, which is very rare in reality. Since the dataset does not give specific information about how Oldpeak is calculated, then we choose to keep those negative value to ensure data integrity. Other numerical variables look normal.

## Distribution of Numerical Covariates

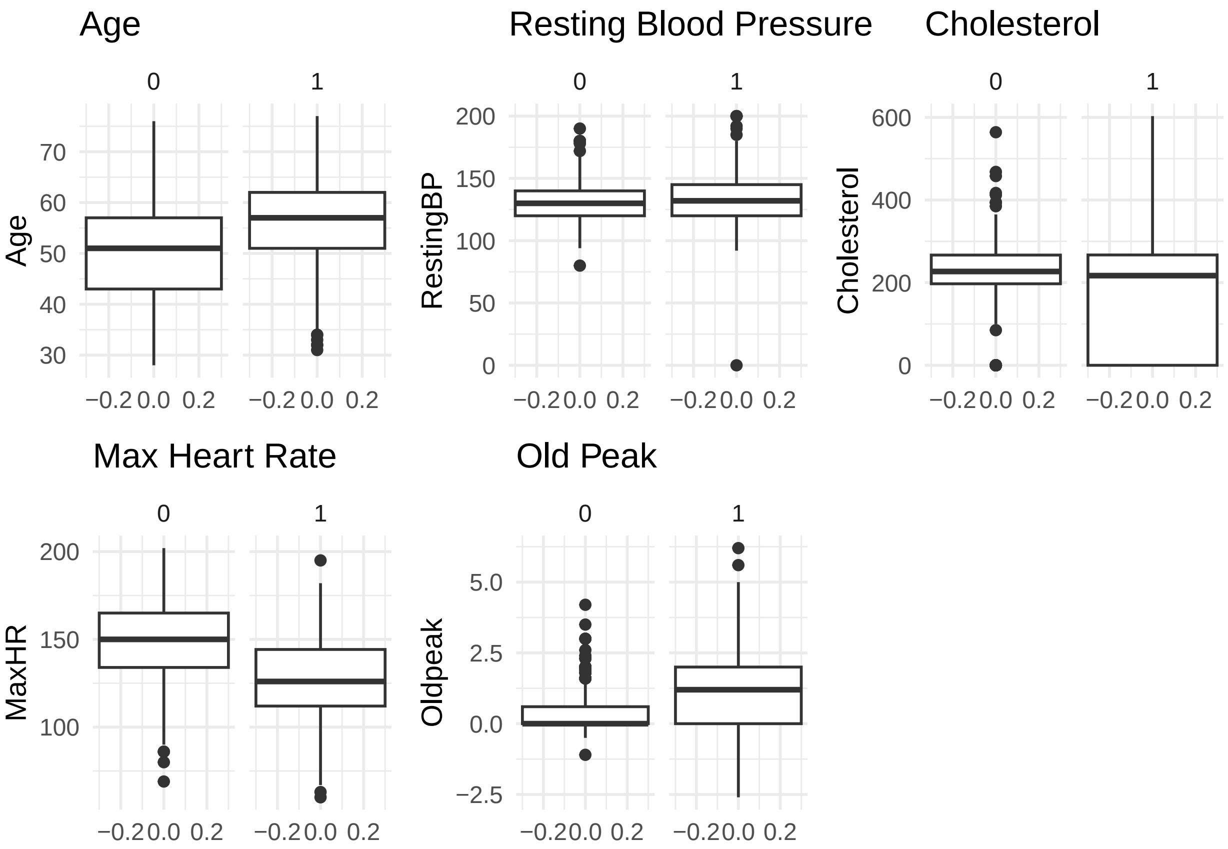
```
# Histogram for numerical variables
num_vars <- sapply(dat, is.numeric)
hist1<-ggplot(dat, aes(x = Age)) + geom_histogram(bins = 30 , alpha = 0.7) + labs(title = "Age")
hist2<-ggplot(dat, aes(x = RestingBP)) + geom_histogram(bins = 30, , alpha = 0.7) + labs(title = "Restin
hist3<-ggplot(dat, aes(x = Cholesterol)) + geom_histogram(bins = 30 , alpha = 0.7) + labs(title = "Chole
hist4<-ggplot(dat, aes(x = MaxHR)) + geom_histogram(bins = 30, , alpha = 0.7) + labs(title = "Max Heart
hist5<-ggplot(dat, aes(x = Oldpeak)) + geom_histogram(bins = 30, , alpha = 0.7) + labs(title = "Old Peak

ggarrange(hist1,hist2,hist3,hist4,hist5, ncol = 3, nrow = 2)
```



The distribution for most variables appear to be approximately normal. For cholesterol level, there are many zeros, which may be problematic. However, it uses units mm/dl instead of mg/dl, which is not commonly used. Since there is no clarification for this unit, then we choose to keep these values.

### Boxplots of Numerical covariates by heart disease

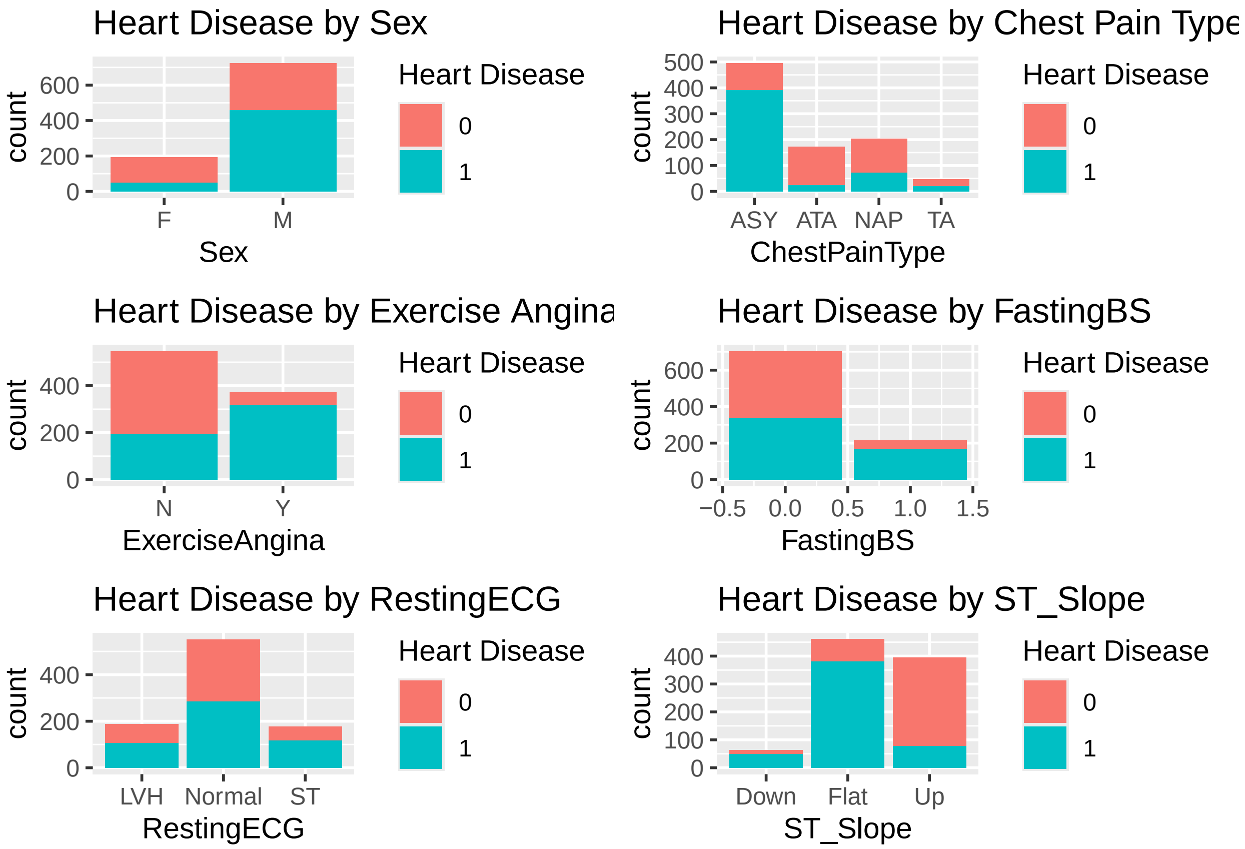
```
num_vars <- sapply(dat, is.numeric)

bx1<-ggplot(dat, aes(y = Age)) + geom_boxplot(width = 0.6) + facet_wrap(~HeartDisease)+labs(title = "Ag

bx2<-ggplot(dat, aes(y = RestingBP)) + geom_boxplot(width = 0.6) + facet_wrap(~HeartDisease)+labs(title

bx3<-ggplot(dat, aes(y = Cholesterol)) + geom_boxplot(width = 0.6) + facet_wrap(~HeartDisease)+labs(titl

bx4<-ggplot(dat, aes(y = MaxHR)) + geom_boxplot(width = 0.6) + facet_wrap(~HeartDisease)+labs(title = "

bx5<-ggplot(dat, aes(y = Oldpeak)) + geom_boxplot(width = 0.6) + facet_wrap(~HeartDisease)+labs(title =


ggarrange(bx1,bx2,bx3,bx4, bx5, ncol = 3, nrow = 2)
```



People who have heart diseases have higher age, lower maximum heart rate, higher old peak, and lower cholesterol. There are some outliers for resting blood Pressure, cholesterol and old peak.

### Relationship between Categorical Covariates and Heart Disease

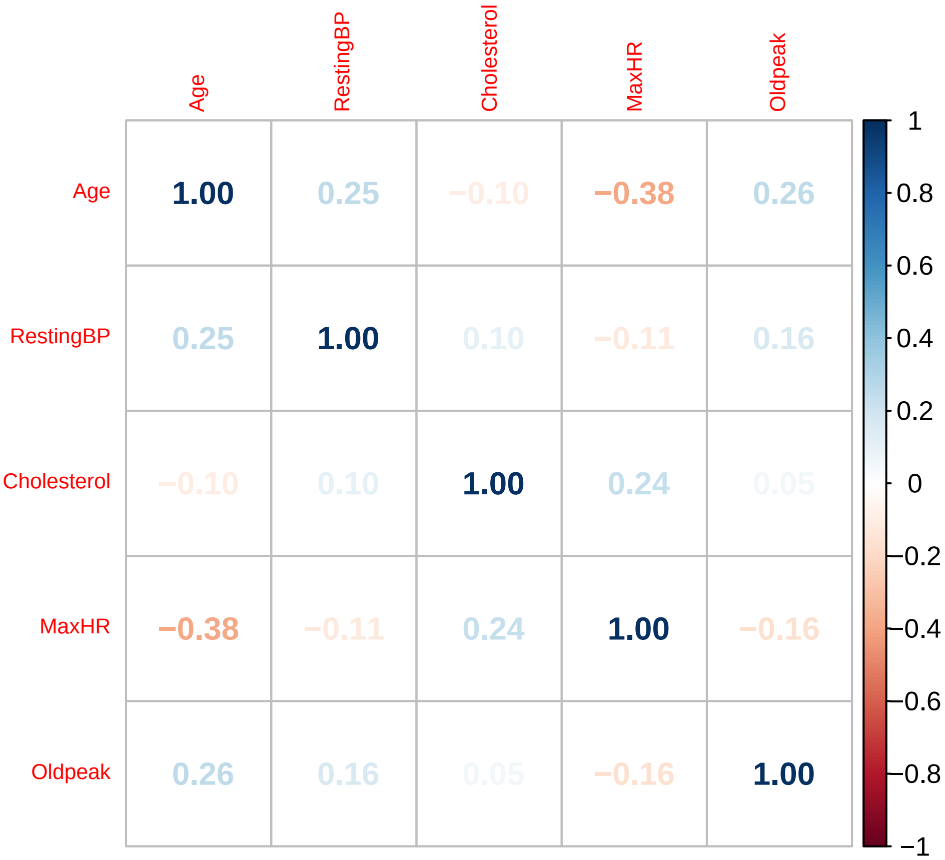
```
# Bar plot for categorical variables
bar1<-ggplot(dat, aes(x = Sex, fill = factor(HeartDisease))) + geom_bar() + labs(title = "Heart Disease
bar2<-ggplot(dat, aes(x = ChestPainType, fill = factor(HeartDisease))) + geom_bar() + labs(title = "Hea
bar3<-ggplot(dat, aes(x = ExerciseAngina, fill = factor(HeartDisease))) + geom_bar() + labs(title = "He
bar4<-ggplot(dat, aes(x = FastingBS, fill = factor(HeartDisease))) + geom_bar() + labs(title = "Heart D
bar5<-ggplot(dat, aes(x = RestingECG, fill = factor(HeartDisease))) + geom_bar() + labs(title = "Heart 
bar6<-ggplot(dat, aes(x = ST_Slope, fill = factor(HeartDisease))) + geom_bar() + labs(title = "Heart Di
ggarrange(bar1,bar2,bar3,bar4,bar5,bar6, ncol = 2, nrow = 3)
```



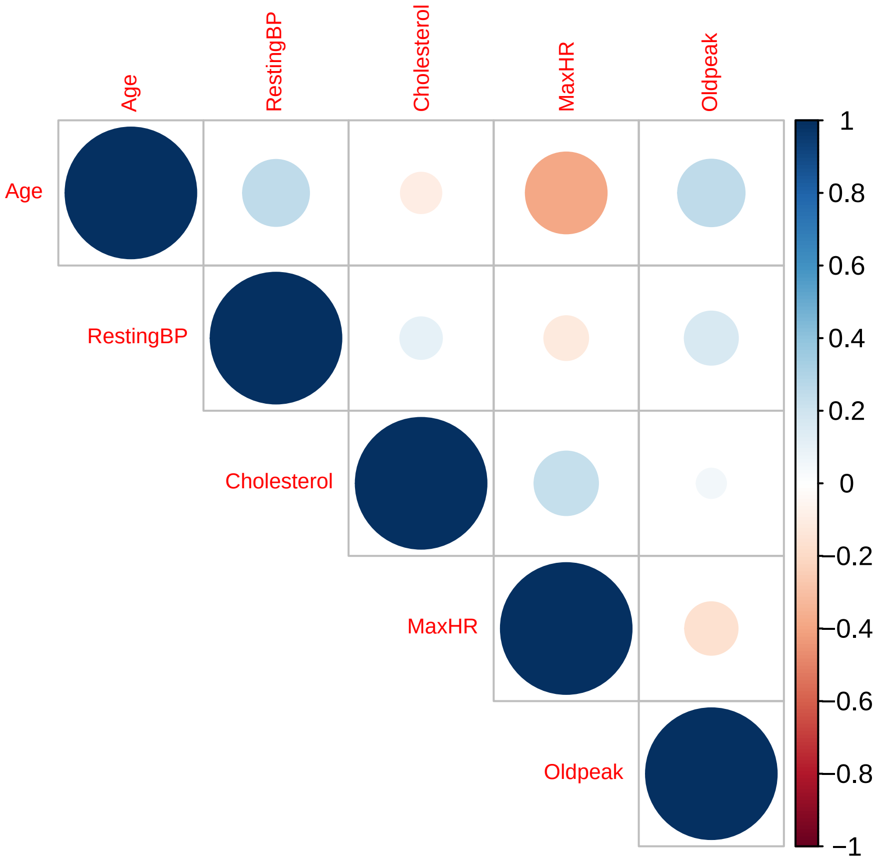
The proportion of male patients with heart disease is greater than proportion of female patients. The probability of heart disease significantly differ by chest pain type, where type ASY: Asymptomatic most likely leads to heart disease, while type ATA: Atypical Angina least likely results in heart disease. Exercise-induced angina also increases the probability of getting heart disease sharply. Patient with fasting blood sugar larger than 120 mg/dl have higher probability of getting heart disease. For resting electrocardiogram results, the proportion of patients with heart disease and without heart disease seems equal for these three types. Patients who have upsloping slope of the peak exercise ST segment have lower probability of getting heart disease.

### Correlation visualization

```
cor_matrix <- cor(dat[num_vars][, c(1:3, 5:6)])

corrplot(cor_matrix, method="number", tl.cex = 0.7)
```



```r
corrplot(cor_matrix, method = "circle", type = "upper", tl.cex = 0.7)
```



The age has moderately negatively correlated with maximum heart rate. The age also has positively correlated with Resting Blood Pressure and Old Peak. The Cholesterol has positively correlated with Maximum Heart Rate.

## Statistical Models and Methodology

### Sequential Log-Likelihood Ratio Tests

We use sequential log-likelihood ratio tests to test nested models starting from the model with all parameters and choose the variable with the highest p-value (most insignificant variable) to perform log-likelihood ratio tests for nested models. If the p-value is larger than 0.05, then the model with that variable excluded is better. This procedure is repeated until the model is not reducible. We follow this procedure to get the best model for each link (logit, probit and complementary log-log).

#### Logit Link

```
dat$Sex <- as.factor(dat$Sex)
dat$ChestPainType <- as.factor(dat$ChestPainType)
dat$RestingECG <- as.factor(dat$RestingECG)
dat$ExerciseAngina <- as.factor(dat$ExerciseAngina)
dat$ST_Slope <- as.factor(dat$ST_Slope)

# Full model with all predictors
full_model <- glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + RestingECG + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "logit"))
```

```r
p_values <- summary(full_model)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingECGNormal | p-value: 0.515021919301189"
```

```r
deviance_full<-summary(full_model)$deviance

# Remove RestingECGNormal

model_1<-glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "logit"))

lr_test_1 <- anova(model_1, full_model, test = "Chisq")
cat("Comparison of model 1 and full model:\n")
```

```
## Comparison of model 1 and full model:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_1$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.6552061
```

```r
cat("p-value: ", lr_test_1$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.720649
```

```r
p_values <- summary(model_1)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingBP | p-value: 0.504585009221759"
```

```r
deviance_1<-summary(model_1)$deviance


# Remove RestingBP

model_2<-glm(HeartDisease ~ Age + Sex + ChestPainType + Cholesterol +
                  FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "logit"))

lr_test_2 <- anova(model_2, model_1, test = "Chisq")
cat("Comparison of model 2 and model 1:\n")
```

```
## Comparison of model 2 and model 1:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_2$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.4436157
```

```r
cat("p-value: ", lr_test_2$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.5053824
```

```r
p_values <- summary(model_2)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
```

```r
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: MaxHR | p-value: 0.468876489199628"
```

```r
deviance_2<-summary(model_2)$deviance


# Remove MaxHR

model_3<-glm(HeartDisease ~ Age + Sex + ChestPainType + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "logit"))

lr_test_3 <- anova(model_3, model_2, test = "Chisq")
cat("Comparison of model 3 and model 2:\n")
```

```
## Comparison of model 3 and model 2:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_3$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.5244608
```

```r
cat("p-value: ", lr_test_3$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.468945
```

```r
p_values <- summary(model_3)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: Age | p-value: 0.0517701795228619"
```

```r
deviance_3<-summary(model_3)$deviance


# Remove Age

model_4<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "logit"))

lr_test_4 <- anova(model_4, model_3, test = "Chisq")
cat("Comparison of model 4 and model 3:\n")
```

```
## Comparison of model 4 and model 3:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_4$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  3.800442
```

```r
cat("p-value: ", lr_test_4$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.05123907
```

```r
p_values <- summary(model_4)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
```

```
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: ST_SlopeUp | p-value: 0.0168401873690736"
```

```
deviance_4<-summary(model_4)$deviance

# Try removing ST_Slope

model_5<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak,
                  data = dat, family = binomial(link = "logit"))

lr_test_5 <- anova(model_5, model_4, test = "Chisq")
cat("Comparison of model 5 and model 4:\n")
```

```
## Comparison of model 5 and model 4:
```

```
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_5$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  129.7074
```

```
cat("p-value: ", lr_test_5$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  6.829707e-29
```

```
p_values <- summary(model_5)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: ChestPainTypeTA | p-value: 0.00103235199327098"
```

```
deviance_5<-summary(model_5)$deviance

# Fail to remove ST_Slope, model 4 is best model
best_logit_model<-model_4
```

The following table summarizes the sequential log-likelihood ratio test for logit link with variable tested and the p-value from log-likelihood ratio test in each step.

```
library(kableExtra)
logit_table <- data.frame(
  Model = c("Model 1", "Model 2", "Model 3", "Model 4", "Model 5"),
  Variable_Tested = c("RestingECGNormal", "RestingBP", "MaxHR", "Age", "ST_slope"),
  p_value_from_log_likelihood_ratio_test = c(0.7206, 0.5054, 0.4689, 0.0518, '6.83e-29')
)

kable(logit_table, format = "latex", longtable = TRUE, caption = "Model Comparison Results, Logit Link"
  kable_styling(full_width = FALSE, position = "center")
```

Table 6: Model Comparison Results, Logit Link

| Model | Variable_Tested | p_value_from_log_likelihood_ratio_test |
|---|---|---|
| Model 1 | RestingECGNormal | 0.7206 |
| Model 2 | RestingBP | 0.5054 |
| Model 3 | MaxHR | 0.4689 |
| Model 4 | Age | 0.0518 |
| Model 5 | ST_slope | 6.83e-29 |

When comparing model 4 with model 5, the p-value is less than 0.05. Model 4 is not reducible anymore, thus this is the best model using logit link. For logit link, we exclude the variables “RestingECGNormal”, “RestingBP”, “MaxHR”, and “Age” from model.

### Probit Link

```
dat$Sex <- as.factor(dat$Sex)
dat$ChestPainType <- as.factor(dat$ChestPainType)
dat$RestingECG <- as.factor(dat$RestingECG)
dat$ExerciseAngina <- as.factor(dat$ExerciseAngina)
dat$ST_Slope <- as.factor(dat$ST_Slope)

# Full model with all predictors
full_model <- glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + RestingECG + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "probit"))

p_values <- summary(full_model)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingECGNormal | p-value: 0.391899285828253"
```

```
deviance_full<-summary(full_model)$deviance

# Remove RestingECGNormal

model_1<-glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "probit"))

lr_test_1 <- anova(model_1, full_model, test = "Chisq")
cat("Comparison of model 1 and full model:\n")
```

```
## Comparison of model 1 and full model:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_1$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  1.193394
```

```r
cat("p-value: ", lr_test_1$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.5506275
```

```r
p_values <- summary(model_1)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: MaxHR | p-value: 0.47421630950207"
```

```r
deviance_1<-summary(model_1)$deviance

# Remove MaxHR

model_2<-glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "probit"))

lr_test_2 <- anova(model_2, model_1, test = "Chisq")
cat("Comparison of model 2 and model 1:\n")
```

```
## Comparison of model 2 and model 1:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_2$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.5048549
```

```r
cat("p-value: ", lr_test_2$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.4773746
```

```r
p_values <- summary(model_2)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingBP | p-value: 0.339536612240353"
```

```r
deviance_2<-summary(model_2)$deviance

# Remove RestingBP

model_3<-glm(HeartDisease ~ Age + Sex + ChestPainType + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "probit"))

lr_test_3 <- anova(model_3, model_2, test = "Chisq")
cat("Comparison of model 3 and model 2:\n")
```

```
## Comparison of model 3 and model 2:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_3$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.9420181
```

```r
cat("p-value: ", lr_test_3$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.3317594
```

```r
p_values <- summary(model_3)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values)
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: Age | p-value: 0.024443164118183"
```

```r
deviance_3<-summary(model_3)$deviance

# See if Age can be removed

model_4<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                  FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "probit"))

lr_test_4 <- anova(model_4, model_3, test = "Chisq")
cat("Comparison of model 4 and model 3:\n")
```

```
## Comparison of model 4 and model 3:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_4$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  5.048791
```

```r
cat("p-value: ", lr_test_4$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.02464312
```

```r
p_values <- summary(model_4)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values)
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: ST_SlopeUp | p-value: 0.00811551812054334"
```

```r
deviance_4<-summary(model_4)$deviance

# Fail to remove Age, model 3 is best model
best_probit_model<-model_3
```

The following table summarizes the sequential log-likelihood ratio test for probit link with variable tested and the p-value from log-likelihood ratio test in each step.

```
library(kableExtra)
logit_table <- data.frame(
  Model = c("Model 1", "Model 2", "Model 3", "Model 4"),
  Variable_Tested = c("RestingECGNormal", "MaxHR", "RestingBP", "Age"),
  p_value_from_log_likelihood_ratio_test = c(0.5506, 0.4742, 0.3318, 0.0246)
)

kable(logit_table, format = "latex", longtable = TRUE, caption = "Model Comparison Results, Probit Link
  kable_styling(full_width = FALSE, position = "center")
```

Table 7: Model Comparison Results, Probit Link

| Model | Variable_Tested | p_value_from_log_likelihood_ratio_test |
|---|---|---:|
| Model 1 | RestingECGNormal | 0.5506 |
| Model 2 | MaxHR | 0.4742 |
| Model 3 | RestingBP | 0.3318 |
| Model 4 | Age | 0.0246 |

When comparing model 3 with model 4, the p-value is less than 0.05. Model 3 is not reducible anymore, thus this is the best model using probit link. For probit link, we exclude the variables “RestingECGNormal”, “MaxHR”, and “RestingBP” from model.

### Complementary log-log Link

```
dat$Sex <- as.factor(dat$Sex)
dat$ChestPainType <- as.factor(dat$ChestPainType)
dat$RestingECG <- as.factor(dat$RestingECG)
dat$ExerciseAngina <- as.factor(dat$ExerciseAngina)
dat$ST_Slope <- as.factor(dat$ST_Slope)

# Full model with all predictors
full_model <- glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + RestingECG + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "cloglog"))

p_values <- summary(full_model)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingECGNormal | p-value: 0.594071731776312"
```

```
deviance_full<-summary(full_model)$deviance

# Remove RestingECGNormal

model_1<-glm(HeartDisease ~ Age + Sex + ChestPainType + RestingBP + Cholesterol +
                  FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                  data = dat, family = binomial(link = "cloglog"))

lr_test_1 <- anova(model_1, full_model, test = "Chisq")
cat("Comparison of model 1 and full model:\n")
```

```
## Comparison of model 1 and full model:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_1$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.9578541
```

```r
cat("p-value: ", lr_test_1$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.6194477
```

```r
p_values <- summary(model_1)$coefficients[, 4]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: Age | p-value: 0.395852568138676"
```

```r
deviance_1<-summary(model_1)$deviance


# Remove Age

model_2<-glm(HeartDisease ~ Sex + ChestPainType + RestingBP + Cholesterol +
                   FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                   data = dat, family = binomial(link = "cloglog"))

lr_test_2 <- anova(model_2, model_1, test = "Chisq")
cat("Comparison of model 2 and model 1:\n")
```

```
## Comparison of model 2 and model 1:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_2$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  0.6856822
```

```r
cat("p-value: ", lr_test_2$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.4076369
```

```r
p_values <- summary(model_2)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: RestingBP | p-value: 0.202465346420105"
```

```r
deviance_2<-summary(model_2)$deviance


# Remove RestingBP

model_3<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                   FastingBS + MaxHR + ExerciseAngina + Oldpeak + ST_Slope,
                   data = dat, family = binomial(link = "cloglog"))

lr_test_3 <- anova(model_3, model_2, test = "Chisq")
cat("Comparison of model 3 and model 2:\n")
```

```
## Comparison of model 3 and model 2:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_3$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  1.75063
```

```r
cat("p-value: ", lr_test_3$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.1857976
```

```r
p_values <- summary(model_3)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: MaxHR | p-value: 0.0640717678346812"
```

```r
deviance_3<-summary(model_3)$deviance


# Remove MaxHR

model_4<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                 FastingBS + ExerciseAngina + Oldpeak + ST_Slope,
                 data = dat, family = binomial(link = "cloglog"))

lr_test_4 <- anova(model_4, model_3, test = "Chisq")
cat("Comparison of model 4 and model 3:\n")
```

```
## Comparison of model 4 and model 3:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_4$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  3.40888
```

```r
cat("p-value: ", lr_test_4$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.06484643
```

```r
p_values <- summary(model_4)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: Oldpeak | p-value: 0.00167453360520785"
```

```r
deviance_4<-summary(model_4)$deviance

# Try removing OldPeak

model_5<-glm(HeartDisease ~ Sex + ChestPainType + Cholesterol +
                 FastingBS + ExerciseAngina + ST_Slope,
                 data = dat, family = binomial(link = "cloglog"))

lr_test_5 <- anova(model_5, model_4, test = "Chisq")
cat("Comparison of model 5 and model 4:\n")
```

```
## Comparison of model 5 and model 4:
```

```r
cat("Log-Likelihood Ratio/Deviance Test statistic: ", lr_test_5$Deviance[2], "\n")
```

```
## Log-Likelihood Ratio/Deviance Test statistic:  9.063929
```

```
cat("p-value: ", lr_test_5$`Pr(>Chi)`[2], "\n")
```

```
## p-value:  0.002607012
```

```
p_values <- summary(model_5)$coefficients[, 4][-1]
predictor_to_remove <- names(p_values)[which.max(p_values)]
max_p_value <- max(p_values[-1])
paste("Next covariate to remove:", predictor_to_remove, "| p-value:", max_p_value)
```

```
## [1] "Next covariate to remove: ST_SlopeFlat | p-value: 0.00521615245371466"
```

```
deviance_5<-summary(model_5)$deviance

# Fail to remove OldPeak, best model is model_4
best_cloglog_model<-model_4
```

The following table summarizes the sequential log-likelihood ratio test for complementary log-log link with variable tested and the p-value from log-likelihood ratio test in each step.

```r
library(kableExtra)
logit_table <- data.frame(
  Model = c("Model 1", "Model 2", "Model 3", "Model 4", "Model 5"),
  Variable_Tested = c("RestingECGNormal", "Age", "RestingBP", "MaxHR", "OldPeak"),
  p_value_from_log_likelihood_ratio_test = c(0.6194, 0.4076, 0.1858, 0.0648, 0.0026)
)

kable(logit_table, format = "latex", longtable = TRUE, caption = "Model Comparison Results, C-log-log Li
  kable_styling(full_width = FALSE, position = "center")
```

Table 8: Model Comparison Results, C-log-log Link

| Model | Variable_Tested | p_value_from_log_likelihood_ratio_test |
|---|---|---:|
| Model 1 | RestingECGNormal | 0.6194 |
| Model 2 | Age | 0.4076 |
| Model 3 | RestingBP | 0.1858 |
| Model 4 | MaxHR | 0.0648 |
| Model 5 | OldPeak | 0.0026 |

When comparing model 4 with model 5, the p-value is less than 0.05. Model 4 is not reducible anymore, thus this is the best model using complementary log-log link. For complementary log-log link, we exclude the variables “RestingECGNormal”, “Age”, “RestingBP”, “MaxHR”, and “OldPeak” from model.

### Comparing the best logit, probit and complementary log-log model

```
paste('Formula of best logit model')
```

```
## [1] "Formula of best logit model"
```

```
best_logit_model$formula
```

```
## HeartDisease ~ Sex + ChestPainType + Cholesterol + FastingBS +
##     ExerciseAngina + Oldpeak + ST_Slope
```

```
paste('Formula of best probit model')
```

```
## [1] "Formula of best probit model"
```

```
best_probit_model$formula
```

```
## HeartDisease ~ Age + Sex + ChestPainType + Cholesterol + FastingBS +
##     ExerciseAngina + Oldpeak + ST_Slope
```

```
paste('Formula of best cloglog model')
```

```
## [1] "Formula of best cloglog model"
```

```
best_cloglog_model$formula
```

```
## HeartDisease ~ Sex + ChestPainType + Cholesterol + FastingBS +
##     ExerciseAngina + Oldpeak + ST_Slope
```

```
paste('DF of best logit model:',best_logit_model$df.residual)
```

```
## [1] "DF of best logit model: 907"
```

```
paste('DF of best probit model:',best_probit_model$df.residual)
```

```
## [1] "DF of best probit model: 906"
```

```
paste('DF of best cloglog model:',best_cloglog_model$df.residual)
```

```
## [1] "DF of best cloglog model: 907"
```

```
paste('Deviance of best logit model',summary(best_logit_model)$deviance)
```

```
## [1] "Deviance of best logit model 599.608808660421"
```

```
paste('Deviance of best probit model',summary(best_probit_model)$deviance)
```

```
## [1] "Deviance of best probit model 596.272789858924"
```

```
paste('Deviance of best cloglog model',summary(best_cloglog_model)$deviance)
```

```
## [1] "Deviance of best cloglog model 618.333892599234"
```

The follow table summarizes which variables were excluded from model equation, number of variables and parameters for each link.

```
compare_1 <- data.frame(
  `Link_Function` = c("Logit", "Probit", "C-log-log"),
  `Variables_Excluded` = c("RestingECGNormal, RestingBP, MaxHR, Age",
                           "RestingECGNormal, RestingBP, MaxHR",
                           "RestingECGNormal, RestingBP, MaxHR, Age"),
  `Number_of_Variables` = c(7, 8, 7),
  `Number_of_Parameters` = c(11, 12, 11)
)

kable(compare_1, format = "latex", longtable = TRUE, caption = "Model Summary for 3 Link Functions") %>
  kable_styling(full_width = FALSE, position = "center")
```

Table 9: Model Summary for 3 Link Functions

| Link_Function | Variables_Excluded | Number_of_Variables | Number_of_Parameters |
|---|---|---:|---:|
| Logit | RestingECGNormal, RestingBP, MaxHR, Age | 7 | 11 |
| Probit | RestingECGNormal, RestingBP, MaxHR | 8 | 12 |
| C-log-log | RestingECGNormal, RestingBP, MaxHR, Age | 7 | 11 |

The follow table summarizes deviance and residual degrees of freedom (equivalent to number of observations minus number of parameters), AIC and BIC for each link.

```
compare_2 <- data.frame(
  `Link_Function` = c("Logit", "Probit", "C-log-log"),
  Deviance = c(599.61, 596.27, 618.33),
  `Residual_DF` = c(907, 907, 907),
  AIC = c(621.61, 620.27, 640.33),
  BIC = c(674.65, 678.14, 693.38)
)

kable(compare_2, format = "latex", longtable = TRUE, caption = "Model Statistics for 3 Link Functions")
  kable_styling(full_width = FALSE, position = "center")
```

Table 10: Model Statistics for 3 Link Functions

| Link_Function | Deviance | Residual_DF | AIC | BIC |
|---|---:|---:|---:|---:|
| Logit | 599.61 | 907 | 621.61 | 674.65 |
| Probit | 596.27 | 907 | 620.27 | 678.14 |
| C-log-log | 618.33 | 907 | 640.33 | 693.38 |

We compare the best model for each link. The probit model has the smallest deviance, but it has one more parameter (age) than logit and c-log-log model (the residual degree of freedom of probit model is 1 less than logit and c-log-log model). And since the deviance of logit model is just marginally greater than probit model, the prediction power of the two models are roughly similar. However, parameters in the logit model are the most interpretable as we can interpret the coefficients in terms of change in odds ratios, thus we prefer the logistic model for result analysis and interpretation. The complementary log-log model has relatively high deviance compared to logit and probit models.

For AIC and BIC, probit model has the smallest AIC while the logit model has the smallest BIC. This is because the probit model has one more parameter than logit model and BIC penalizes it more significantly.

# Model Diagnostics and Jusitification

We perform model diagnostics for both logit and probit link since their prediction power is similar based on their deviance and degree of freedom of residuals.

## Goodness of fit

### Ungrouped Pearson residuals

```r
pearson_logit <- residuals(best_logit_model, type = "pearson")
chisq_logit <- sum(pearson_logit^2)
p_value <- 1 - pchisq(chisq_logit, best_logit_model$df.residual)

print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.536289404731893"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the logit link model fits the data well.

```r
pearson_probit <- residuals(best_probit_model, type = "pearson")
chisq_probit <- sum(pearson_probit^2)
p_value <- 1 - pchisq(chisq_probit, best_probit_model$df.residual)

print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.894133261445169"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the probit link model fits the data well.

```r
pearson_probit <- residuals(best_cloglog_model, type = "pearson")
chisq_probit <- sum(pearson_probit^2)
p_value <- 1 - pchisq(chisq_probit, best_cloglog_model$df.residual)

print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.16061231177685"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the cloglog link model fits the data well.

### Grouped Pearson residuals(Success vs. Failure)

```r
actual_success <- sum(dat$HeartDisease)
actual_failure <- 918 - actual_success
pred_probs<-fitted(best_logit_model)
pred_success<-sum(pred_probs > 0.5)
pred_failure<-918 - pred_success
Pearson_stat<-(actual_success-pred_success)^2/pred_success +
  (actual_failure-pred_failure)^2/pred_failure
p_Pearson<-pchisq(q = Pearson_stat, df = 1, lower.tail = FALSE)
print(paste("p-value:", p_Pearson))
```

```
## [1] "p-value: 0.286012719360935"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the logit link model fits the data well.

```r
pred_probs<-fitted(best_probit_model)
pred_success<-sum(pred_probs > 0.5)
pred_failure<-918 - pred_success
Pearson_stat<-(actual_success-pred_success)^2/pred_success +
  (actual_failure-pred_failure)^2/pred_failure
p_Pearson<-pchisq(q = Pearson_stat, df = 1, lower.tail = FALSE)
print(paste("p-value:", p_Pearson))
```

```
## [1] "p-value: 0.317351506448646"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the probit link model fits the data well.

```
pred_probs<-fitted(best_cloglog_model)
pred_success<-sum(pred_probs > 0.5)
pred_failure<-918 - pred_success
Pearson_stat<-(actual_success-pred_success)^2/pred_success +
  (actual_failure-pred_failure)^2/pred_failure
p_Pearson<-pchisq(q = Pearson_stat, df = 1, lower.tail = FALSE)
print(paste("p-value:", p_Pearson))
```

```
## [1] "p-value: 0.550936535782802"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the cloglog link model fits the data well.

### Hosmer–Lemeshow (g=10)

For this dataset, some explanatory variables are continuous, so we can use Hosmer–Lemeshow Test.

```
dat_logit<-dat

dat_logit$est_prob <- predict(best_logit_model, type = "response")


dat_logit$group <- cut(dat_logit$est_prob,
                 breaks = quantile(dat_logit$est_prob, probs = seq(0, 1, 0.1)),
                 include.lowest = TRUE)

gof <- dat_logit %>%
  group_by(group) %>%
  summarise(
    observed = sum(HeartDisease),
    total = n(),
    expected = sum(est_prob)
  ) %>%
  mutate(stat = (observed - expected)^2 / expected)

statistic <- sum(gof$stat)
df <- nrow(gof) - 2
p_value <- pchisq(statistic, df = df, lower.tail = FALSE)


print(paste("Chi-squared statistic:", statistic))
```

```
## [1] "Chi-squared statistic: 6.1903632718506"
```

```
print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.625917784107009"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the logit link model fits the data well.

```r
dat_probit<-dat

dat_probit$est_prob <- predict(best_probit_model, type = "response")


dat_probit$group <- cut(dat_probit$est_prob,
                 breaks = quantile(dat_probit$est_prob, probs = seq(0, 1, 0.1)),
                 include.lowest = TRUE)

gof2 <- dat_probit %>%
  group_by(group) %>%
  summarise(
    observed = sum(HeartDisease),
    total = n(),
    expected = sum(est_prob)
  ) %>%
  mutate(stat = (observed - expected)^2 / expected)

statistic <- sum(gof2$stat)
df <- nrow(gof2) - 2
p_value <- pchisq(statistic, df = df, lower.tail = FALSE)


print(paste("Chi-squared statistic:", statistic))
```

```
## [1] "Chi-squared statistic: 5.48121780848267"
```

```r
print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.705120004193544"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the probit link model fits the data well.

```r
dat_cloglog<-dat

dat_cloglog$est_prob <- predict(best_cloglog_model, type = "response")


dat_cloglog$group <- cut(dat_cloglog$est_prob,
                 breaks = quantile(dat_cloglog$est_prob, probs = seq(0, 1, 0.1)),
                 include.lowest = TRUE)

gof2 <- dat_cloglog %>%
  group_by(group) %>%
  summarise(
    observed = sum(HeartDisease),
    total = n(),
    expected = sum(est_prob)
  ) %>%
  mutate(stat = (observed - expected)^2 / expected)

statistic <- sum(gof2$stat)
```

```r
df <- nrow(gof2) - 2
p_value <- pchisq(statistic, df = df, lower.tail = FALSE)


print(paste("Chi-squared statistic:", statistic))
```

```
## [1] "Chi-squared statistic: 9.16360773806022"
```

```r
print(paste("p-value:", p_value))
```

```
## [1] "p-value: 0.328682868379845"
```

Since the p-value is larger than 0.05, we fail to reject null hypothesis. Thus, the cloglog link model fits the data well.

## Model comparison

### Deviance

Deviance is used to compare nested models, so it is not appropriate for logit and probit link model.

### Log-Likelihood ratio test

Log-likelihood ratio chi-squared is used to compare nested models, so it is not appropriate for comparion of models in different link functions (logit, probit and complementary log-log).

### AIC/BIC

```r
AIC(best_logit_model, best_probit_model,best_cloglog_model)
```

```
##                    df      AIC
## best_logit_model   11 621.6088
## best_probit_model  12 620.2728
## best_cloglog_model 11 640.3339
```

```r
BIC(best_logit_model, best_probit_model,best_cloglog_model)
```

```
##                    df      BIC
## best_logit_model   11 674.6530
## best_probit_model  12 678.1392
## best_cloglog_model 11 693.3781
```

AIC suggests probit-link model is better, but BIC suggest logit-link model is better.

# Interpretation of Results

1. **Results Overview**:
   - We performed Pearson chi-squared, Hosmer–Lemeshow and AIC/BIC to compare goodness of fit of logit-link model, Probit-link model and Complementary Log-log Link model
2. **Hypotheses**:
   - $$H_0 :$$

     The model's predicted probabilities match observed outcomes (good fit).
   - $$H_1 :$$

     The model's predicted probabilities deviate from observations (poor fit).

3. **Key Results**:
    - Logit-link model:
        - Pearson chi-squared: p-value: 0.536
        - Hosmer–Lemeshow: Chi-squared statistic: 6.190, p-value: 0.626
        - DF: 11
        - AIC: 621.6088, BIC: 674.6530

    - Probit-link model:
        - Pearson chi-squared: p-value: 0.894
        - Hosmer–Lemeshow: Chi-squared statistic: 5.481, p-value: 0.705
        - DF: 12
        - AIC: 620.2728, BIC: 678.1392
    - Complementary Log-log Link model:
        - Pearson chi-squared: p-value: 0.161
        - Hosmer–Lemeshow: Chi-squared statistic: 9.164, p-value: 0.329
        - DF: 11
        - AIC: 640.3339, BIC: 693.3781
4. **Statistical Conclusion**:
   With $p > 0.05$, we fail to reject H0 for both Logit-link model, Probit-link model and Complementary Log-log Link model, suggesting no significant evidence of model misfit.

## Parameter Estimation, Confidence Intervals and Estimated Odds Ratio

We use the logistic regression or logit link for this part, since logistic regression gives the best interpretability.

```
summary(best_logit_model)
```

```
##
## Call:
## glm(formula = HeartDisease ~ Sex + ChestPainType + Cholesterol +
##     FastingBS + ExerciseAngina + Oldpeak + ST_Slope, family = binomial(link = "logit"),
##     data = dat)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -0.481859   0.562252  -0.857 0.391436
## SexM              1.454586   0.278086   5.231 1.69e-07 ***
## ChestPainTypeATA -1.878771   0.322002  -5.835 5.39e-09 ***
## ChestPainTypeNAP -1.706720   0.260758  -6.545 5.94e-11 ***
## ChestPainTypeTA  -1.458703   0.424979  -3.432 0.000598 ***
## Cholesterol      -0.004124   0.001026  -4.019 5.84e-05 ***
## FastingBS         1.193157   0.271642   4.392 1.12e-05 ***
## ExerciseAnginaY   0.991359   0.235370   4.212 2.53e-05 ***
## Oldpeak           0.410094   0.115694   3.545 0.000393 ***
## ST_SlopeFlat      1.443532   0.425675   3.391 0.000696 ***
## ST_SlopeUp       -1.060365   0.443634  -2.390 0.016840 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1262.14  on 917  degrees of freedom
## Residual deviance:  599.61  on 907  degrees of freedom
## AIC: 621.61
##
```

```
## Number of Fisher Scoring iterations: 5
```

```
# Calculate OR and confidence interval
OR <- exp(coef(best_logit_model))
OR_CI <- exp(confint(best_logit_model))
```

```
## Waiting for profiling to be done...
```

```
# Merge results
results <- data.frame(
  OR = OR,
  Lower_CI = OR_CI[, 1],
  Upper_CI = OR_CI[, 2]
)
print(results)
```

```
##                          OR   Lower_CI  Upper_CI
## (Intercept)      0.6176343 0.20736467 1.8886997
## SexM             4.2827104 2.50196503 7.4544504
## ChestPainTypeATA 0.1527778 0.07985661 0.2831571
## ChestPainTypeNAP 0.1814600 0.10796524 0.3006203
## ChestPainTypeTA  0.2325376 0.10024453 0.5330257
## Cholesterol      0.9958842 0.99384652 0.9978607
## FastingBS        3.2974766 1.95407599 5.6782873
## ExerciseAnginaY  2.6948946 1.70074019 4.2855921
## Oldpeak          1.5069590 1.20472576 1.8974607
## ST_SlopeFlat     4.2356306 1.80328579 9.6324508
## ST_SlopeUp       0.3463293 0.14192160 0.8126061
```

# Conclusions

1. **Model Selection and Fit Assessment**:

   - Variable Selection: Through sequential log-likelihood ratio tests, the optimal predictors for heart disease were identified as: -Logit/Cloglog Models: Sex, ChestPainType, Cholesterol, FastingBS, ExerciseAngina, Oldpeak, ST_Slope. -Probit Model: Additionally included Age.
   - Goodness-of-Fit: All models passed Pearson chi-square and Hosmer-Lemeshow tests (p-values > 0.05), indicating no significant lack of fit: -Logit: Chi-squared = 6.19 (p=0.626) -Probit: Chi-squared = 5.48 (p=0.705) -Cloglog: Chi-squared = 9.16 (p=0.329)

2. **Model Performance Comparison**:
   -Deviance/AIC/BIC: -Probit showed the lowest deviance (596.27) and AIC (620.27), suggesting marginally better fit. -Logit had the lowest BIC (674.65), and one less parameter than Probit, more parsimonious for prediction. -Cloglog performed slightly worse but remained viable.

3. **Key Predictors of Heart Disease**: -Strong Positive Association: -SexM (Male): OR = 4.28 (logit). -ExerciseAnginaY (Yes): OR = 2.69. -ST_SlopeFlat (vs. Down): OR = 4.23. -Strong Negative Association: -Cholesterol: Higher levels reduced risk OR = 0.996 (logit coef = -0.004). -ChestPainTypeATA/NAP/TA (vs. typical angina): All reduced risk.

4. **Limitations**: -Limitations: -Unmeasured parameter (e.g., lifestyle factors) may affect estimates.

5. **Final Conclusions**: -The probit model has higher precision. If high precision is the goal, the probit model is recommended. However, it is more complex than the logit model and requires one more parameters. In comparison, the cloglog model does not provide any advantages. Taking all factors into consideration, the logit model has broader applicability in this analysis.

# Reference

fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved [Date Retrieved] from https://www.kaggle.com/fedesoriano/heart-failure-prediction.