University of San Carlos
DCISM

Final Culminating Activity

MATH 3109

PREPARED BY:
Woodrow A. Fajardo

SUBMITTED TO:
Prof. Elmer S. Poliquit

MATH-3109
DECEMBER 2025

Contents

Raw Data based from Kaggle:

1 Introduction

Diabetes is a significant public health concern that demands accurate methods for early detection. Predictive modeling allows us to identify individuals at high risk before severe complications arise. This study utilizes the Pima Indians Diabetes Database to construct robust statistical models. We aim to determine which diagnostic measurements most accurately predict the onset of diabetes and to model the non-linear fluctuations of insulin levels using applied regression techniques.

1.1 Project Objective

This project aims to predict the onset of diabetes based on diagnostic measures. We will also model insulin levels using regression techniques. The goal is to identify key risk factors and understand non-linear relationships in the medical data.

1.2 Literature Review

Nassiwa and Zeng identify Glucose and BMI as the primary predictors of diabetes. This study seeks to replicate those findings using the Pima Indians dataset. We hypothesize that metabolic markers will outrank demographic factors like age in predictive power.

1.3 Raw Data

The dataset consists of 768 observations from the Pima Indians Diabetes Database. It includes medical predictor variables and one target variable, Outcome. Initial inspection reveals missing values encoded as zeros, which requires correction before analysis.

R | SETTING UP ENVIRONMENT

```r
library(mice)
library(corrplot)
library(ggplot2)
library(splines)
library(performance)
library(see)
library(ggpubr)

setwd("~/School/MATH-3109/FINAL-1")
raw_data <- read.csv("raw_dataset.csv")
View(raw_data)
head(raw_data)
```

RAW DATA HEAD

```
Pregnancies Glucose BloodPressure SkinThickness Insulin
          6     148            72            35       0
          1      85            66            29       0
          8     183            64             0       0
          1      89            66            23      94
          0     137            40            35     168
          5     116            74             0       0

  BMI  DiabetesPedigreeFunction  Age   Outcome
  33.6                    0.627  50       1
  26.6                    0.351  31       0
  23.3                    0.672  32       1
  28.1                    0.167  21       0
  43.1                    2.288  33       1
  25.6                    0.201  30       0
```

2 Data Cleaning

Real-world medical data is rarely perfect. We must first address biologically impossible zero values that represent missing data to avoid biased results.

2.1 Converting Zeroes Values to NA

The summary confirms substantial missingness disguised as zeros.

R | GETTING COLUMNS WITH ZERO VALUES

```r
as.matrix(colSums(raw_data == 0))
```

COLUMNS AND NUMBER OF ZERO VALES

```
  Pregnancies                111
  Glucose                      5
  BloodPressure               35
  SkinThickness              227
  Insulin                    374
  BMI                         11
  DiabetesPedigreeFunction     0
  Age                          0
  Outcome                    500
```

R | CONVERTING ZERO VALUES TO NA

```r
col_w_zeroes <- c("Pregnancies", "Glucose", "BloodPressure", "SkinThickness",
                  "Insulin", "BMI")
raw_data[col_w_zeroes] <- lapply(raw_data[col_w_zeroes], function(x)
                                 ifelse(x == 0, NA, x))

summary(raw_data)
```

VISUALIZING SUMMARY ACCOUNTING NA's

```
  Pregnancies        Glucose      BloodPressure    SkinThickness      Insulin
 Min.   : 1.000   Min.   : 44.0   Min.   : 24.00   Min.   : 7.00   Min.   : 14.00
 1st Qu.: 2.000   1st Qu.: 99.0   1st Qu.: 64.00   1st Qu.:22.00   1st Qu.: 76.25
 Median : 4.000   Median :117.0   Median : 72.00   Median :29.00   Median :125.00
 Mean   : 4.495   Mean   :121.7   Mean   : 72.41   Mean   :29.15   Mean   :155.55
 3rd Qu.: 7.000   3rd Qu.:141.0   3rd Qu.: 80.00   3rd Qu.:36.00   3rd Qu.:190.00
 Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00   Max.   :846.00
 NA's   :111      NA's   :5       NA's   :35       NA's   :227     NA's   :374

  BMI        DiabetesPedigreeFunction      Age            Outcome
 Min.   :18.20   Min.   :0.0780          Min.   :21.00   Min.   :0.000
 1st Qu.:27.50   1st Qu.:0.2437          1st Qu.:24.00   1st Qu.:0.000
 Median :32.30   Median :0.3725          Median :29.00   Median :0.000
 Mean   :32.46   Mean   :0.4719          Mean   :33.24   Mean   :0.349
 3rd Qu.:36.60   3rd Qu.:0.6262          3rd Qu.:41.00   3rd Qu.:1.000
 Max.   :67.10   Max.   :2.4200          Max.   :81.00   Max.   :1.000
 NA's   :11
```

Insulin has the highest missing rate, with 374 zero entries. Variables like Glucose, Blood Pressure, and BMI also contain invalid zeros that require correction.

2.2 MICE Imputation

Dropping rows with missing data would lose too much information. We will use Multivariate Imputation by Chained Equations (MICE) with predictive mean matching to estimate and fill these missing values.

R | MICE IMPUTATION PMM as METHOD

```r
init <- mice(raw_data, maxit = 0)
methods <- init$method #$
methods[col_w_zeroes] <- "pmm"
imputed_data <- mice(raw_data, method = methods, m = 5, maxit = 5, seed = 123)
final_data <- complete(imputed_data, 1)
head(final_data)
```

View full R script output here.

R | CHECK FINAL DATA CONTAINS NA's

```r
anyNA(final_data)
```

RETURN TRUE CONTAINS NA's OTHERWISE NONE

```
[1] FALSE
```

2.3 Imputation Validation

We must verify that our statistical imputation did not distort the natural distribution of the data. Visual inspection confirms the integrity of the data. The imputed distributions (blue) closely mirror the original distributions (red). This suggests the MICE algorithm preserved the underlying data structure without introducing bias.

Refer to Appendix Section for the full R script.

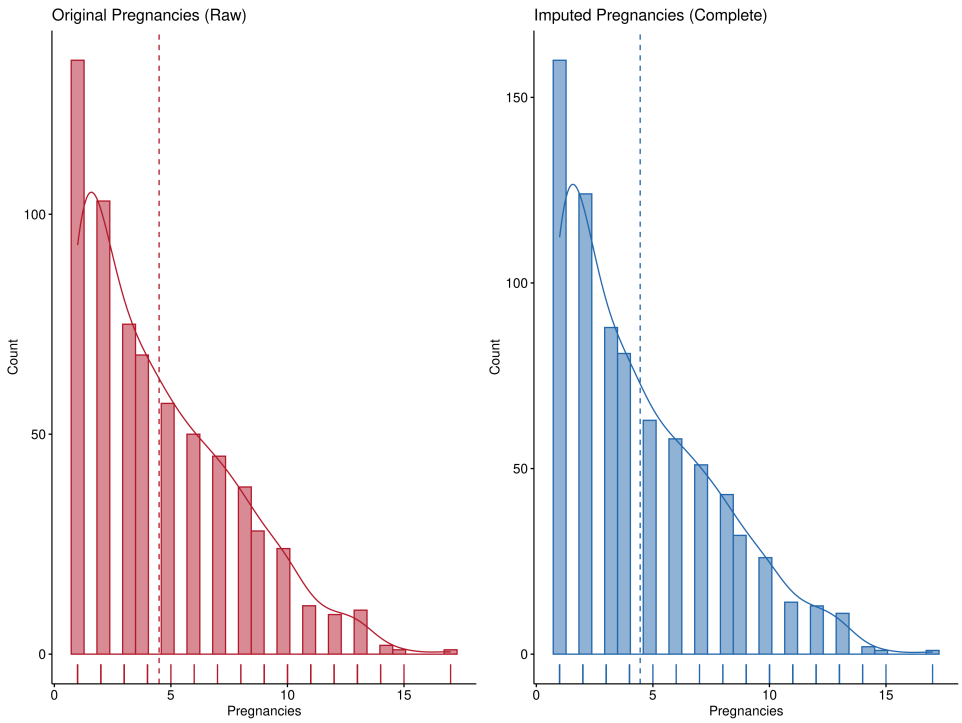

FIGURE 1
Before and After Imputation (Pregnancies).

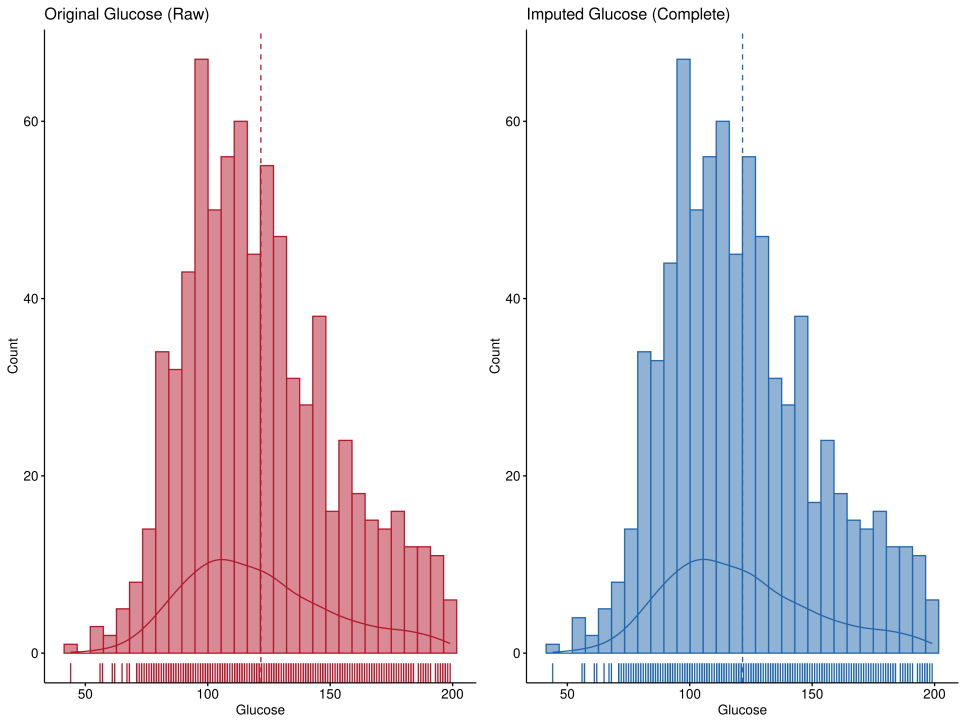

FIGURE 2
Before and After Imputation (Glucose).

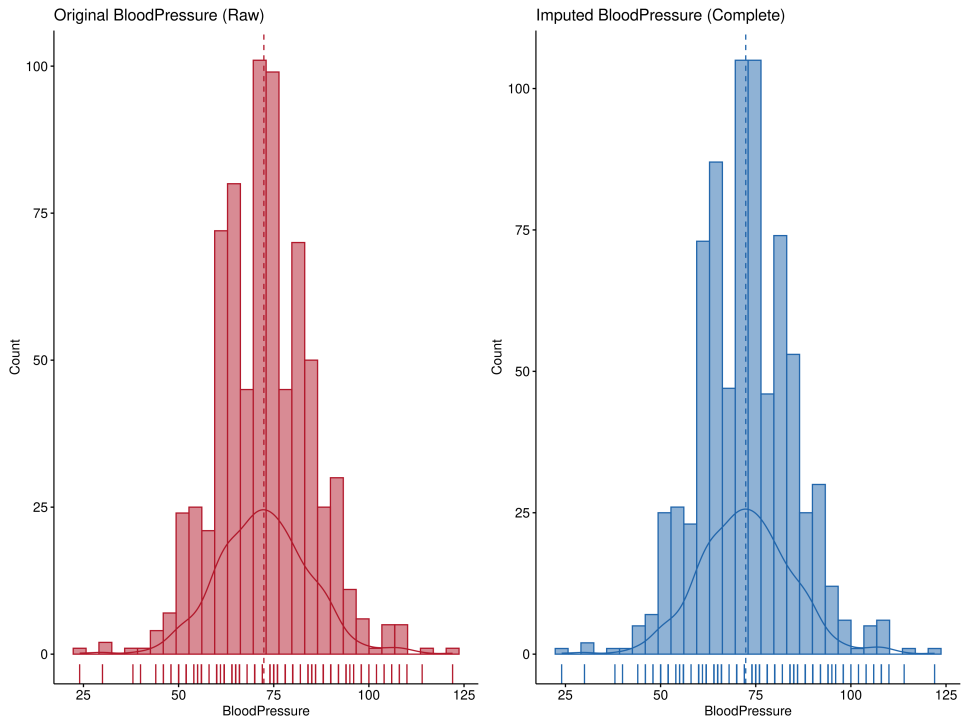

FIGURE 3
Before and After Imputation (BloodPressure).

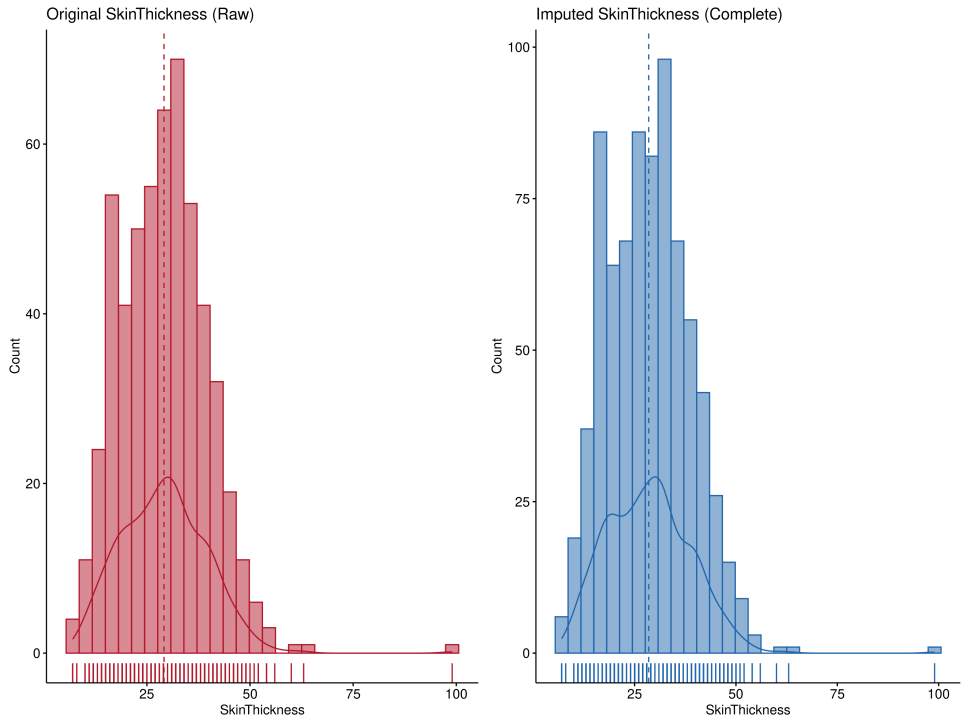

FIGURE 4
Before and After Imputation (SkinThickness).

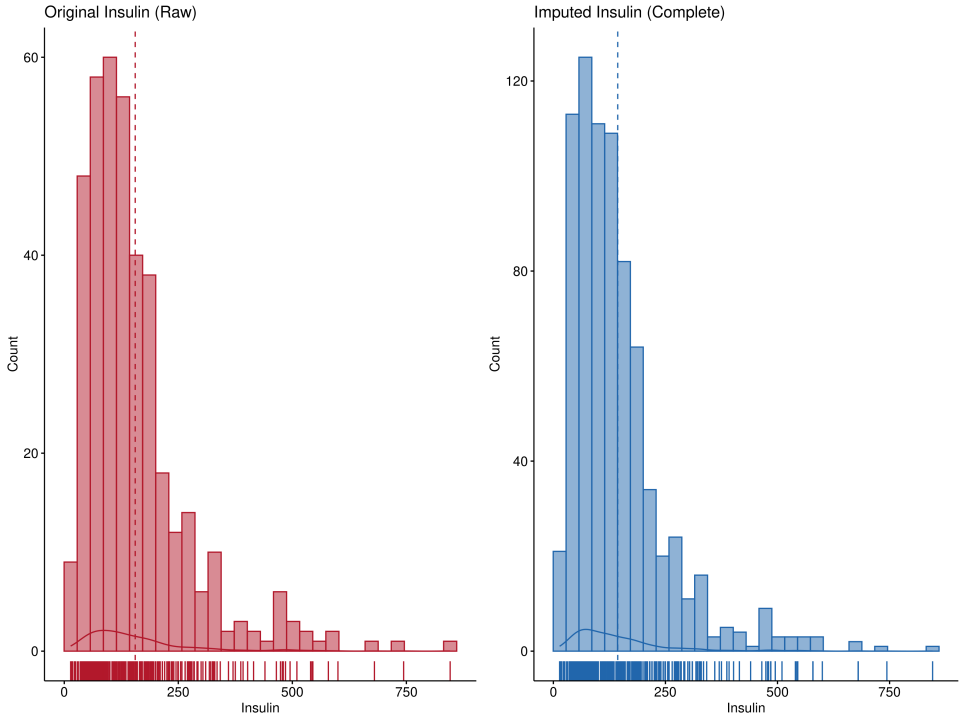

FIGURE 5
Before and After Imputation (Insulin).

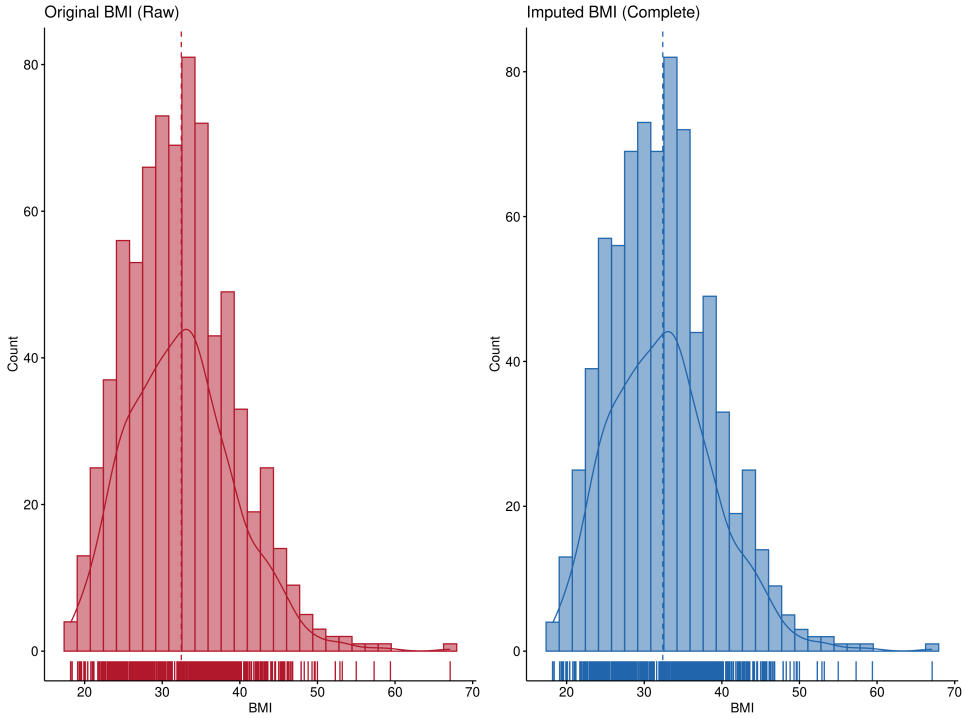

FIGURE 6
Before and After Imputation (BMI).

3 Exploratory Analysis

Before modeling, we examine the relationships between variables to identify potential predictors. We specifically check for multicollinearity to ensure our regression models remain stable and interpretable.

3.1 Correlation Matrix

The matrix reveals strong correlations between `Glucose` and the diabetes `Outcome`. We also observe multicollinearity between `Age` and `Pregnancies`. This validates our selection of predictors but suggests caution regarding variance inflation in the models.

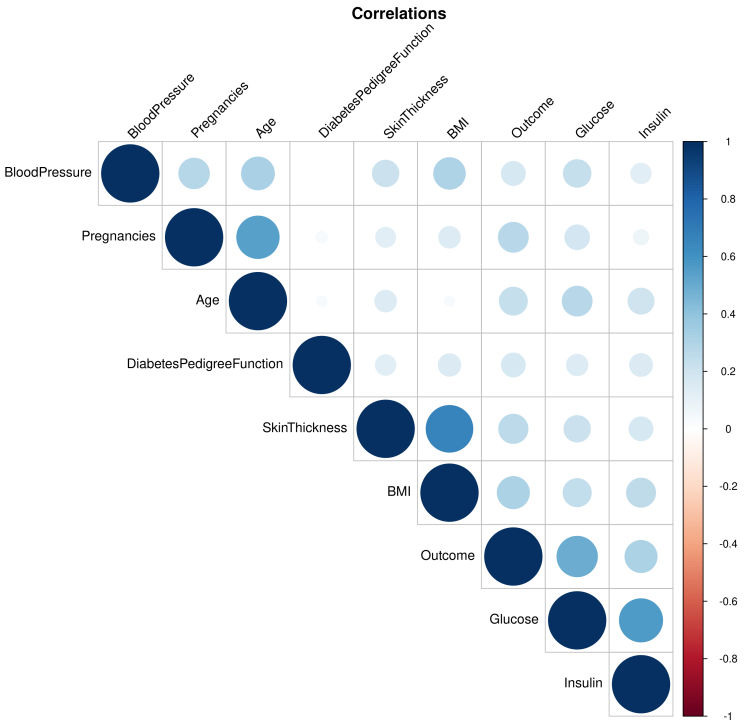

FIGURE 7
Correlation Matrix Plot

R | CORRELATION MATRIX PLOT

```r
cor_matrix <- cor(final_data, use = "complete.obs")
corrplot(cor_matrix, method = "circle", type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45, title = "Correlations", mar=c(0,0,1,0))
```

3.2 Analyzing Linearity

We test the linearity assumption for our continuous regression model by examining the relationship between Age and Insulin. The local regression plot shows a distinct curve. Insulin levels rise in early adulthood but plateau after age 50. A simple linear line cannot capture this, justifying the need for non-linear modeling techniques.

4 Modelling Logistic Regression

We fit a logistic regression model to predict the binary diabetes `Outcome` using our cleaned dataset. The model separates diabetic and non-diabetic patients based on the selected risk factors.

4.1 Model Specification

`Glucose`, `BMI`, and `Pregnancies` are highly significant predictors ($p < 0.001$). However, `Age` is not statistically significant ($p = 0.337$). This is likely due to its strong correlation with `Pregnancies`, which absorbed the predictive power in this model.

R | MODELLING LOGISTIC REGRESSION

```r
set.seed(123)
sample_index <- sample(1:nrow(final_data), 0.8 * nrow(final_data))
train_data <- final_data[sample_index, ]
test_data <- final_data[-sample_index, ]

log_model <- glm(Outcome ~ Glucose + BMI + Age + Pregnancies +
                 DiabetesPedigreeFunction, data = train_data,
                 family= "binomial")

summary(log_model)
```

SUMMARY LOG-RES MODEL

```
glm(formula = Outcome ~ Glucose + BMI + Age + Pregnancies +
                        DiabetesPedigreeFunction, family = "binomial",
                        data = train_data)

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -9.200623   0.809675 -11.363  < 2e-16 ***
Glucose                   0.037413   0.004099   9.128  < 2e-16 ***
BMI                       0.079814   0.016719   4.774 1.81e-06 ***
Age                       0.009594   0.010003   0.959 0.337506
Pregnancies               0.130485   0.037653   3.465 0.000529 ***
DiabetesPedigreeFunction  0.659625   0.321580   2.051 0.040247 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 796.42  on 613  degrees of freedom
Residual deviance: 568.82  on 608  degrees of freedom
AIC: 580.82

Number of Fisher Scoring iterations: 5
```

4.2 Odds Ratio Interpretation

We convert the model coefficients into odds ratios to quantify the biological risk in human-readable terms. For every one-unit increase in BMI, the odds of diabetes rise by roughly 8.3%. Glucose shows a similar positive risk. The Diabetes Pedigree Function has the highest ratio, nearly doubling the odds for each unit increase.

R | ODDS RATIO

```r
as.matrix(exp(coef(log_model)))
```

ODDS RATIO OUTPUT

```
(Intercept)              0.0001009765
Glucose                  1.0381215630
BMI                      1.0830855110
Age                      1.0096399484
Pregnancies              1.1393803946
DiabetesPedigreeFunction 1.9340669910
```

4.3 Model Performance

We assess the model's goodness-of-fit by examining the deviance and AIC values. The model significantly reduces deviance from 796 to 568. This drop indicates that our selected predictors explain a substantial amount of the variation in diabetes onset.

5 Linear Regression (Predict Insulin)

We shift to a continuous model to predict Insulin levels based on Glucose, BMI, and Age. This analysis aims to capture the factors driving insulin fluctuation.

R | MODELLING LINEAR REGRESSION

```r
linear_simple <- lm(Insulin ~ Glucose + BMI + Age,
                    data = final_data)
summary(linear_simple)
```

LINEAR REGRESS OUTPUT

```
Call:
lm(formula = Insulin ~ Glucose + BMI + Age, data = final_data)

Residuals:
    Min      1Q  Median      3Q     Max
-289.02  -45.91  -14.70   25.01  567.44

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -167.6456    19.1120  -8.772  < 2e-16 ***
Glucose        1.8543     0.1125  16.481  < 2e-16 ***
BMI            2.0651     0.4791   4.310 1.84e-05 ***
Age            0.5693     0.2841   2.004   0.0455 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 89.01 on 764 degrees of freedom
Multiple R-squared:  0.3401,    Adjusted R-squared:  0.3375
F-statistic: 131.2 on 3 and 764 DF,  p-value: < 2.2e-16
```

The simple linear model explains 34% of the variation in insulin levels. Glucose and BMI are the strongest predictors ($p < 0.001$). Age is statistically significant but has a weaker effect ($p = 0.046$). The model suggests that for every year a person ages, their insulin increases by about 0.57 units.

R | LINEAR SPLINE MODEL

```r
linear_spline <- lm (Insulin ~ Glucose + BMI + bs(Age, degree = 3),
                     data = final_data)
summary(linear_spline)
```

LINEAR SPLINE MODEL OUTPUT

```
Call:
lm(formula = Insulin ~ Glucose + BMI + bs(Age, degree = 3), data = final_data)

Residuals:
    Min      1Q  Median      3Q     Max
-299.47  -47.78  -14.42   26.00  554.90

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          -151.3071    18.2230  -8.303 4.61e-16 ***
Glucose                 1.8612     0.1123  16.571  < 2e-16 ***
BMI                     2.2633     0.4871   4.646 3.98e-06 ***
bs(Age, degree = 3)1  -57.2504    32.8240  -1.744   0.0815 .
bs(Age, degree = 3)2   80.9994    50.5390   1.603   0.1094
bs(Age, degree = 3)3    5.4944    62.7432   0.088   0.9302
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 88.82 on 762 degrees of freedom
Multiple R-squared:  0.3447,    Adjusted R-squared:  0.3404
F-statistic: 80.16 on 5 and 762 DF,  p-value: < 2.2e-16
```

The B-spline model improves the fit slightly. The R-squared value rises to 34.5%. Glucose and BMI remain the strongest drivers of insulin levels. The spline coefficients capture the complex relationship between age and insulin. The shifting signs in these coefficients confirm the non-linear "rise and fall" pattern seen in our earlier graphs.

5.1 B-Spline Justification

Standard linear regression assumes a constant rate of change. Our EDA showed that aging effects saturate over time. Therefore, we apply a cubic B-spline to Age to model this biological non-linearity.

Refer to Appendix Section for the full R script.

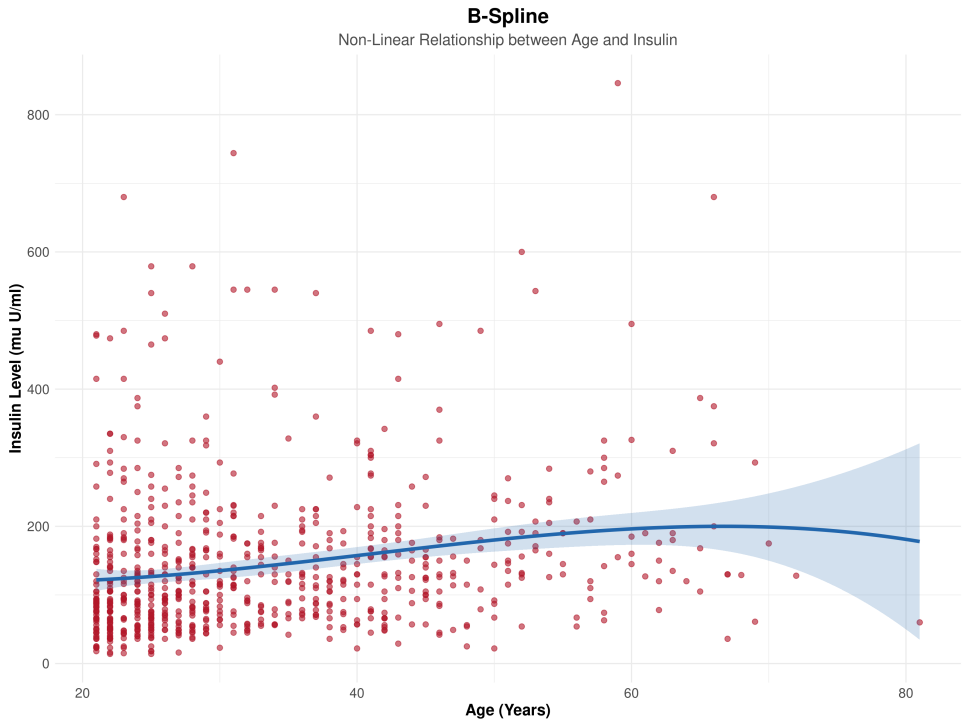

FIGURE 8
B-Spline Plot.

5.2 Comparison Linear vs. B-Spline

We statistically compare the simple linear model against the B-spline model using an ANOVA test. The ANOVA results show a p-value of 0.07. While strictly above the 0.05 threshold, it suggests marginal significance. Given the visual evidence of curvature, we retain the spline model for its biological realism.

R | LINEAR & B-SPLIME COMPARISON

```r
linear_simple <- lm(Insulin ~ Glucose + BMI + Age, data = final_data)
linear_spline <- lm (Insulin ~ Glucose + BMI + bs(Age, degree = 3), data =
  ↪  final_data)
anova(linear_simple, linear_spline)
```

ANOVA OTUPUT

```
Analysis of Variance Table

Model 1: Insulin ~ Glucose + BMI + Age
Model 2: Insulin ~ Glucose + BMI + bs(Age, degree = 3)
  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1    764 6052997
2    762 6010880  2     42116 2.6696 0.06993 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We compared the two models using an ANOVA test. The p-value is 0.07. This result is marginally significant. It is strictly above the standard 0.05 threshold. However, visual evidence shows a clear curve in the data. Therefore, we accept the spline model because it better reflects biological reality.

5.3 Residual Diagnostics

We examine the residuals to ensure the model assumptions of homoscedasticity and normality are met. Diagnostic plots show that the residuals are randomly scattered, satisfying the linearity assumption. Point 460 is flagged as influential but remains within acceptable Cook's distance limits. The model fits the data well.

FIGURE 9
Residual Plot.

R | RESIDUAL PLOT

```r
image_rend <- check_model(linear_spline)
png("splien_model.png", width = 12, height = 9, units = "in", res = 300)
print(image_rend)
dev.off()
```

The diagnostic plots validate the model assumptions. The residuals are randomly distributed around zero. This confirms linearity. The Q-Q plot shows the errors follow a normal distribution. Observation 460 is an outlier but does not distort the model. The variance remains consistent across the data range.

6 Discussions

We synthesize our findings to validate our initial hypotheses. This discussion connects our statistical outputs back to the medical literature and the project objectives.

6.1 Validation of Literature Claims

Our results confirm the claims of Nassiwa and Zeng. `Glucose` and `BMI` were the strongest predictors in our logistic model ($\mathbf{p < 2e - 16}$). This proves that immediate metabolic markers are more predictive than general demographics like age.

6.2 Key Findings

This study produced three major statistical insights. We successfully imputed missing data without bias. Logistic regression identified Glucose, BMI, and Pregnancies as primary risk factors. Finally, we demonstrated that the relationship between Age and Insulin is non-linear, requiring spline modeling.

6.3 Limitations (if any)

The dataset has inherent constraints despite our cleaning efforts. The dataset s relatively small, with only **768** observations. Despite robust imputation, the original missingness in Insulin was high. Future studies would benefit from a larger, more complete dataset to improve precision.

7 Appendix

Appendix Imputation Comparison

HISTOGRAM COMPARISON CODE EXAMPLE

```r
p1 <- gghistogram(raw_data, x = "Pregnancies",
    title = "Original Pregnancies (Raw)",
    xlab = "Pregnancies", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

p2 <- gghistogram(final_data, x = "Pregnancies",
    title = "Imputed Pregnancies (Complete)",
    xlab = "Pregnancies", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)

p1 <- gghistogram(raw_data, x = "Glucose",
    title = "Original Glucose (Raw)",
    xlab = "Glucose", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

p2 <- gghistogram(final_data, x = "Glucose",
    title = "Imputed Glucose (Complete)",
    xlab = "Glucose", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)

p1 <- gghistogram(raw_data, x = "BloodPressure",
    title = "Original BloodPressure (Raw)",
    xlab = "BloodPressure", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)
```

HISTOGRAM COMPARISON CODE EXAMPLE

```
p2 <- gghistogram(final_data, x = "BloodPressure",
    title = "Imputed BloodPressure (Complete)",
    xlab = "BloodPressure", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)

p1 <- gghistogram(raw_data, x = "SkinThickness",
    title = "Original SkinThickness (Raw)",
    xlab = "SkinThickness", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

p2 <- gghistogram(final_data, x = "SkinThickness",
    title = "Imputed SkinThickness (Complete)",
    xlab = "SkinThickness", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)

p1 <- gghistogram(raw_data, x = "Insulin",
    title = "Original Insulin (Raw)",
    xlab = "Insulin", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

p2 <- gghistogram(final_data, x = "Insulin",
    title = "Imputed Insulin (Complete)",
    xlab = "Insulin", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)

p1 <- gghistogram(raw_data, x = "BMI",
    title = "Original BMI (Raw)",
    xlab = "BMI", ylab = "Count",
    fill = "#b2182b",
    color = "#b2182b",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)
```

HISTOGRAM COMPARISON CODE EXAMPLE

```
p2 <- gghistogram(final_data, x = "BMI",
    title = "Imputed BMI (Complete)",
    xlab = "BMI", ylab = "Count",
    fill = "#2166ac",
    color = "#2166ac",
    add = "mean", rug = TRUE, add_density = TRUE,
    ggtheme = theme_pubr()
)

ggarrange(p1, p2, ncol = 2, nrow = 1)
```

7.1 Appendix B-Spline

STATISTICAL MODELING

```
ggplot(final_data, aes(x = Age, y = Insulin)) +
  geom_point(color = "#b2182b", alpha = 0.6, size = 2) +
  geom_smooth(method = "lm",
              formula = y ~ bs(x, degree = 3),
              color = "#2166ac",
              fill = "#2166ac",
              alpha = 0.2,
              size = 1.5) +

  labs(title = "B-Spline",
       subtitle = "Non-Linear Relationship between Age and Insulin",
       x = "Age (Years)",
       y = "Insulin Level (mu U/ml)") +

  theme_minimal() +
  theme(
    plot.title = element_text(face = "bold", hjust = 0.5, size = 18),
    plot.subtitle = element_text(hjust = 0.5, color = "gray30", size = 14),
    axis.title = element_text(face = "bold", size = 14),
    axis.text = element_text(size = 12))
```