Q1

```
# Load libraries
library(readxl)
library(ggplot2)

# Read the Excel file (adjust the path if needed)
data <- read_excel("LabDataG3.xlsx")

# Fit regression models
# Degudent Group
degudent_model <- lm(dE00_Group_D3 ~ Cut_Group_D3, data = data)

# Zirkonzahn Group
zirkonzahn_model <- lm(dE00_Group_Z3 ~ Cut_Group_Z3, data = data)

# Summary of regression for Degudent Group
summary(degudent_model)

# Summary of regression for Zirkonzahn Group
summary(zirkonzahn_model)

# Plot for Degudent Group
ggplot(data, aes(x = Cut_Group_D3, y = dE00_Group_D3)) +
  geom_point() +
  geom_smooth(method = "lm", col = "blue") +
  ggtitle("Degudent Group: Cut Level vs ΔE00") +
  xlab("Cut Level (Degudent)") +
  ylab("ΔE00 (Degudent)") +
  theme_minimal()

# Plot for Zirkonzahn Group
ggplot(data, aes(x = Cut_Group_Z3, y = dE00_Group_Z3)) +
  geom_point() +
  geom_smooth(method = "lm", col = "red") +
  ggtitle("Zirkonzahn Group: Cut Level vs ΔE00") +
  xlab("Cut Level (Zirkonzahn)") +
  ylab("ΔE00 (Zirkonzahn)") +
  theme_minimal()
```

In our regression analysis for both the Degudent and Zirkonzahn groups, we created a linear regression model with Cut Level as the independent variable and ΔE00 as the dependent variable.

Overall, both models fit the data well. Our R-squared values are close to 1 showing that Cut Level is a strong predictor of ΔE00 for both groups. On a side note, the Degudent group has a slightly greater slope, suggesting that ΔE00 increases steeper with Cut Level compared to Zirkonzahn which had a lesser slope.

Degudent:

Model Fit: The R-squared is 0.978, meaning 97.8% of the variation in ΔE00 can be explained by Cut Level.

Coefficients: The intercept is -1.0951, and the slope for Cut Level is 0.0299, indicating a positive relationship between Cut Level and ΔE00.

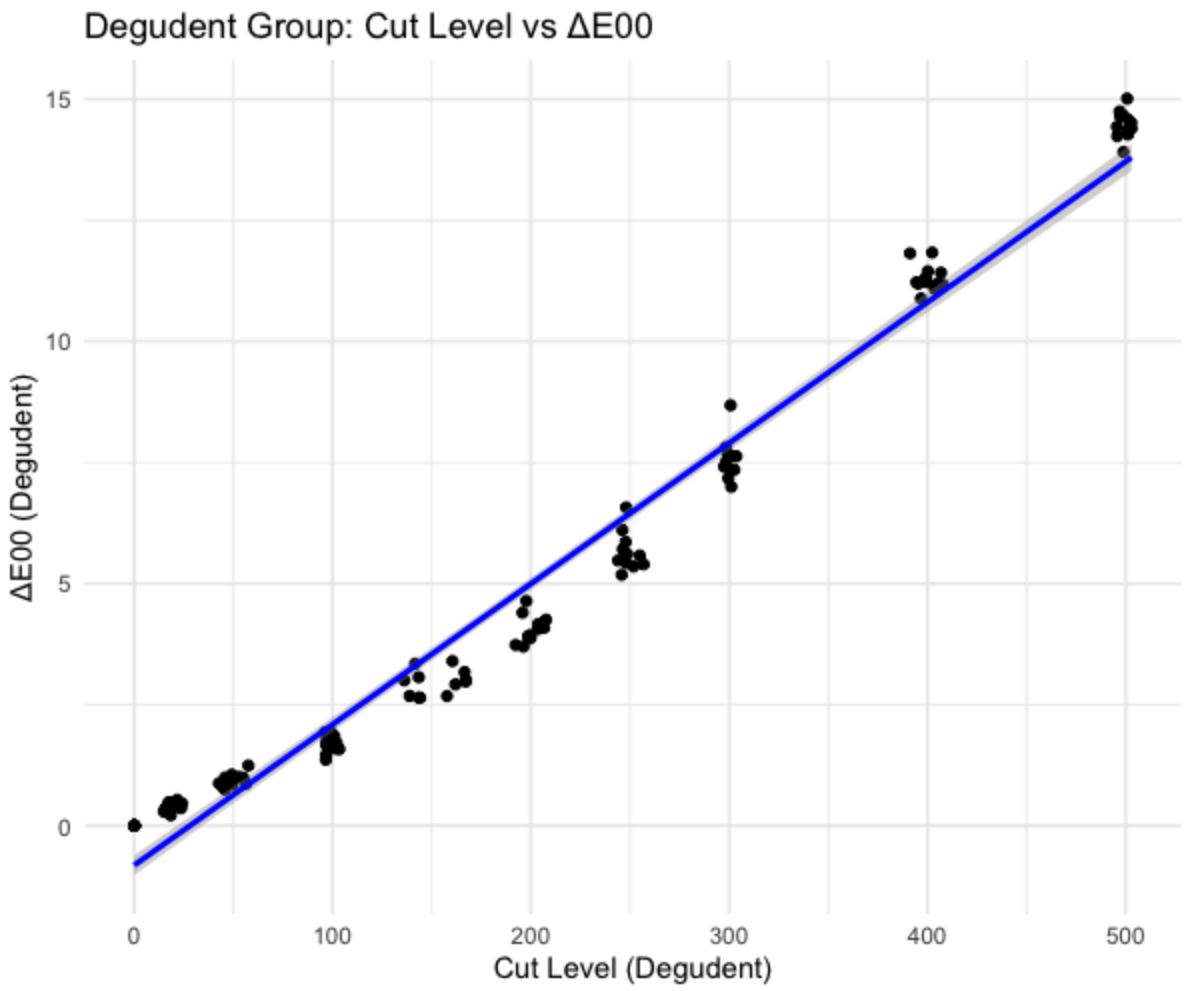
Statistical Significance: The p-value for Cut Level is extremely low, confirming a significant fit.

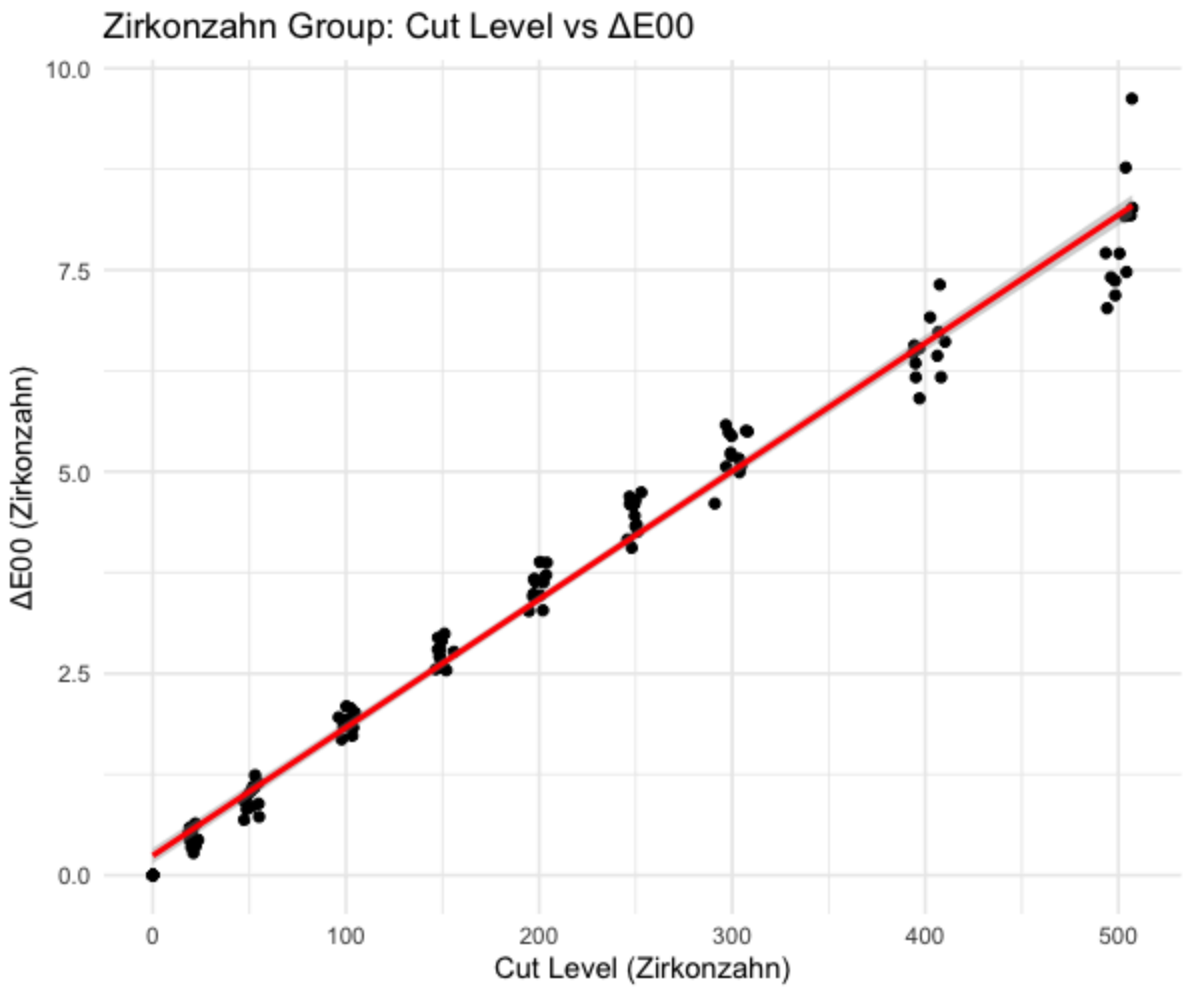
Zirkonzahn:

Model Fit: The R-squared is 0.980, so 98.0% of the variation in ΔE00 is explained by Cut Level.

Coefficients: The intercept is 0.3253, and the slope is 0.0156, also indicating a positive relationship.

Statistical Significance: Similar to Degudent, the p-value is very low, showing a significant fit.


Degudent Group: Cut Level vs ΔE00
ΔE00 (Degudent)
Cut Level (Degudent)
0
5
10
15
0
100
200
300
400
500



Q2

```
# Plot diagnostic plots for Degudent Group
par(mfrow = c(2, 2)) # Arrange 2x2 plot layout for diagnostics
plot(degudent_model, which = 1:4) # Residuals vs Fitted, QQ, Scale-Location, Residuals vs Leverage

# Plot diagnostic plots for Zirkonzahn Group
par(mfrow = c(2, 2)) # Arrange 2x2 plot layout for diagnostics
plot(zirkonzahn_model, which = 1:4) # Residuals vs Fitted, QQ, Scale-Location, Residuals vs Leverage


# Coefficients for Degudent Group
degudent_coeff <- coef(degudent_model)
degudent_cut_estimate <- (1.8 - degudent_coeff[1]) / degudent_coeff[2]
```

```
# Coefficients for Zirkonzahn Group
zirkonzahn_coeff <- coef(zirkonzahn_model)
zirkonzahn_cut_estimate <- (1.8 - zirkonzahn_coeff[1]) / zirkonzahn_coeff[2]

# Display estimates
degudent_cut_estimate
zirkonzahn_cut_estimate

# Define the acceptability threshold for ΔE00
y_h <- 1.8

# Extract coefficients for Degudent Group
degudent_intercept <- coef(degudent_model)[1]
degudent_slope <- coef(degudent_model)[2]
degudent_cut_estimate <- (y_h - degudent_intercept) / degudent_slope

# Extract coefficients for Zirkonzahn Group
zirkonzahn_intercept <- coef(zirkonzahn_model)[1]
zirkonzahn_slope <- coef(zirkonzahn_model)[2]
zirkonzahn_cut_estimate <- (y_h - zirkonzahn_intercept) / zirkonzahn_slope

# Display estimated cut levels
cat("Predicted Cut Level for Degudent at ΔE00 = 1.8:", degudent_cut_estimate, "\n")
cat("Predicted Cut Level for Zirkonzahn at ΔE00 = 1.8:", zirkonzahn_cut_estimate, "\n")
```

Results:

Degudent Cut Estimates Intercept : 90.04147

Zirkonzahn Cut Estimates Intercept: 98.10153

Predicted Cut Level for Degudent at ΔE00 = 1.8: 90.04147

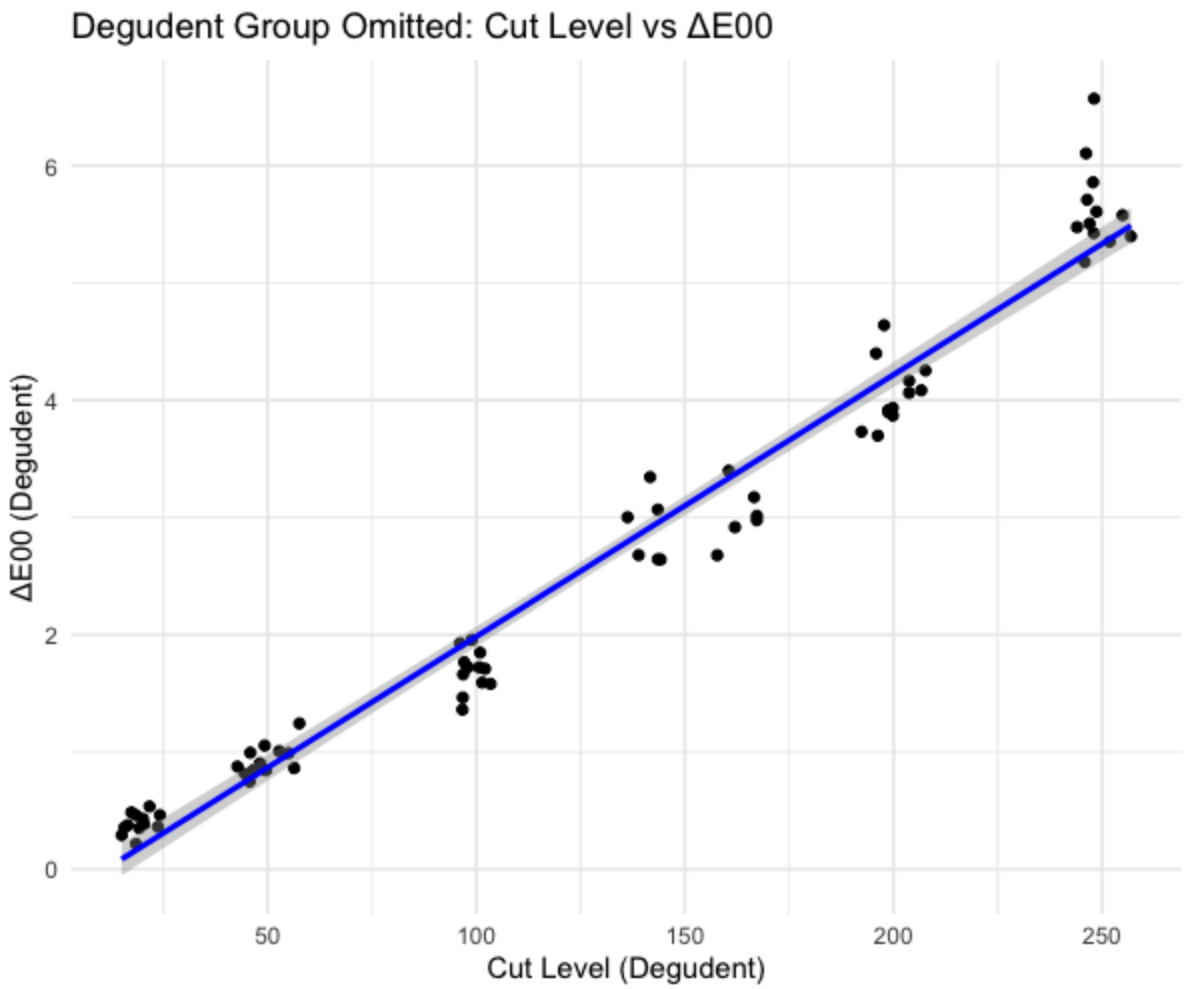
Predicted Cut Level for Zirkonzahn at ΔE00 = 1.8: 98.10153

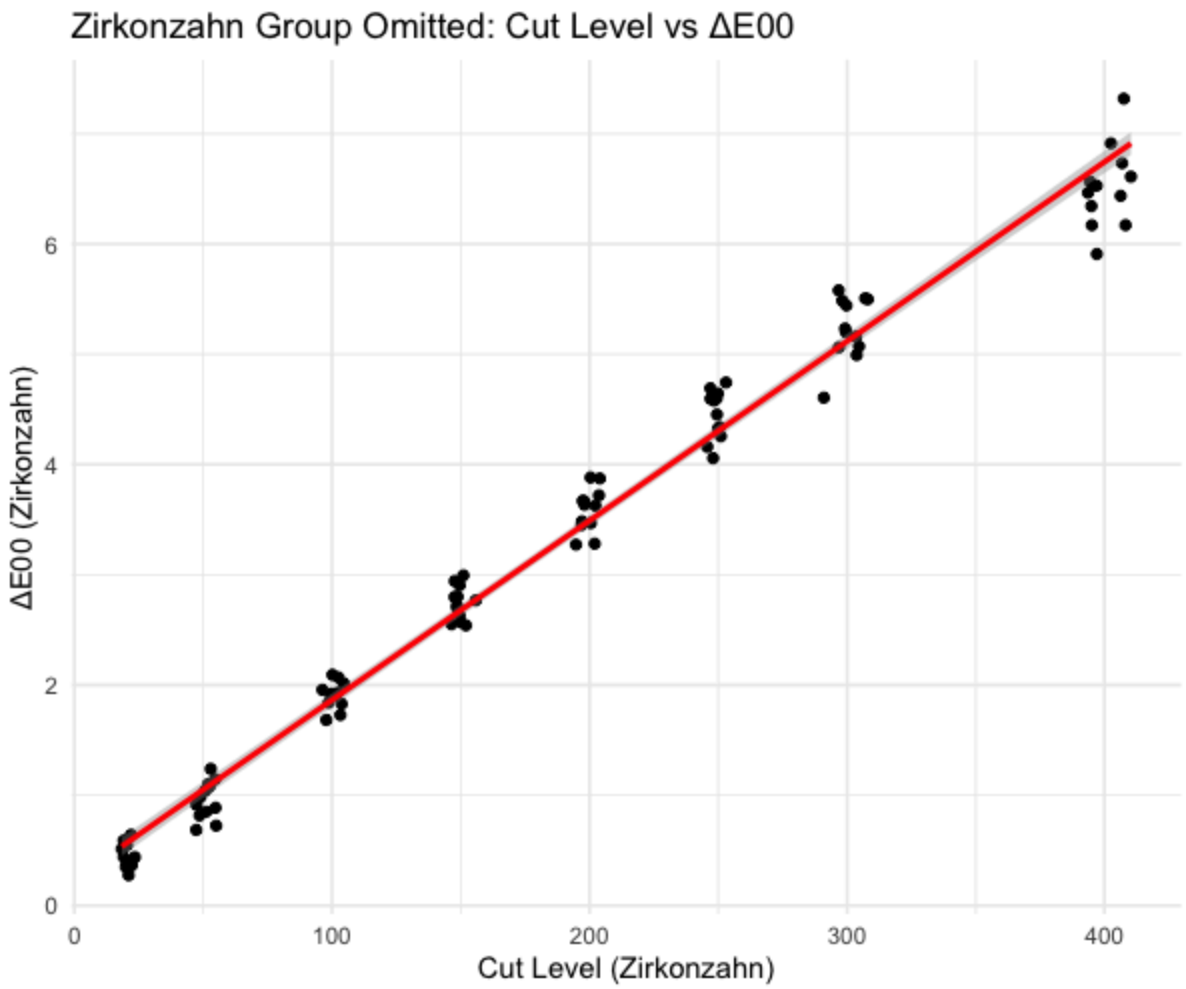
Q3

```
degudent_data_omit <- subset(data, !(`Desired Cut` %in% c(0, 300, 400, 500)))
zirkonzahn_data_omit <- subset(data, !(`Desired Cut` %in% c(0, 500)))

degudent_model_omit <- lm(dE00_Group_D3 ~ Cut_Group_D3, data = degudent_data_omit)
zirkonzahn_model_omit <- lm(dE00_Group_Z3 ~ Cut_Group_Z3, data = zirkonzahn_data_omit)

summary(degudent_model_omit)
```

```
summary(zirkonzahn_model_omit)

d_new_data <- data.frame(`Cut_Group_D3` = 500)
z_new_data <- data.frame(`Cut_Group_Z3` = 500)

degudent_prediction <- predict(degudent_model_omit, d_new_data, interval = "prediction", level
= 0.95)
zirkonzahn_prediction <- predict(zirkonzahn_model_omit, z_new_data, interval = "prediction",
level = 0.95)

degudent_prediction
zirkonzahn_prediction

# Plot for Degudent Group
ggplot(degudent_data_omit, aes(x = Cut_Group_D3, y = dE00_Group_D3)) +
  geom_point() +
  geom_smooth(method = "lm", col = "blue") +
  ggtitle("Degudent Group Omitted: Cut Level vs ΔE00") +
  xlab("Cut Level (Degudent)") +
  ylab("ΔE00 (Degudent)") +
  theme_minimal()

# Plot for Zirkonzahn Group
ggplot(zirkonzahn_data_omit, aes(x = Cut_Group_Z3, y = dE00_Group_Z3)) +
geom_point() +
geom_smooth(method = "lm", col = "red") +
ggtitle("Zirkonzahn Group Omitted: Cut Level vs ΔE00") +
xlab("Cut Level (Zirkonzahn)") +
ylab("ΔE00 (Zirkonzahn)") +
theme_minimal()
```

We found that for the Degudent Group we needed to omit 300, 400, and 500 from our data because they were outside of our initial prediction interval and heavily skewed the data. For Zirkonzahn, we only removed data at the 500 level because that gave us the best R-squared and fit to our data very nicely.


Degudent Group Omitted: Cut Level vs ΔE00
ΔE00 (Degudent)
0
2
4
6
50
100
150
200
250
Cut Level (Degudent)



Q4

```
# Check assumptions for Degudent model
par(mfrow = c(2, 2))
plot(degudent_model_omit)

# Check assumptions for Zirkonzahn model
plot(zirkonzahn_model_omit)
par(mfrow = c(1, 1))

# Degudent prediction with confidence interval
d_new_data <- data.frame(Cut_Group_D3 = 100)
degudent_prediction <- predict(degudent_model_omit, d_new_data, interval = "confidence",
level = 0.95)

# Zirkonzahn prediction with confidence interval
z_new_data <- data.frame(Cut_Group_Z3 = 100)
```

```
zirkonzahn_prediction <- predict(zirkonzahn_model_omit, z_new_data, interval = "confidence",
level = 0.95)

# Display predictions and confidence intervals
degudent_prediction
zirkonzahn_prediction
```

Results:

Cut level ^xh for Degudent is 100

Confidence Interval for ^xh is (1.9001, 2.068229)

Cut level ^xh for Zirkonzahn is 100

Confidence Interval for ^xh is (1.804379, 1.92392)

The confidence interval only has a small overlap, suggesting that there is no significant difference between the two cut levels. If the overlap between the two cut levels was greater then there could be an argument that the difference is significant enough to change the values.