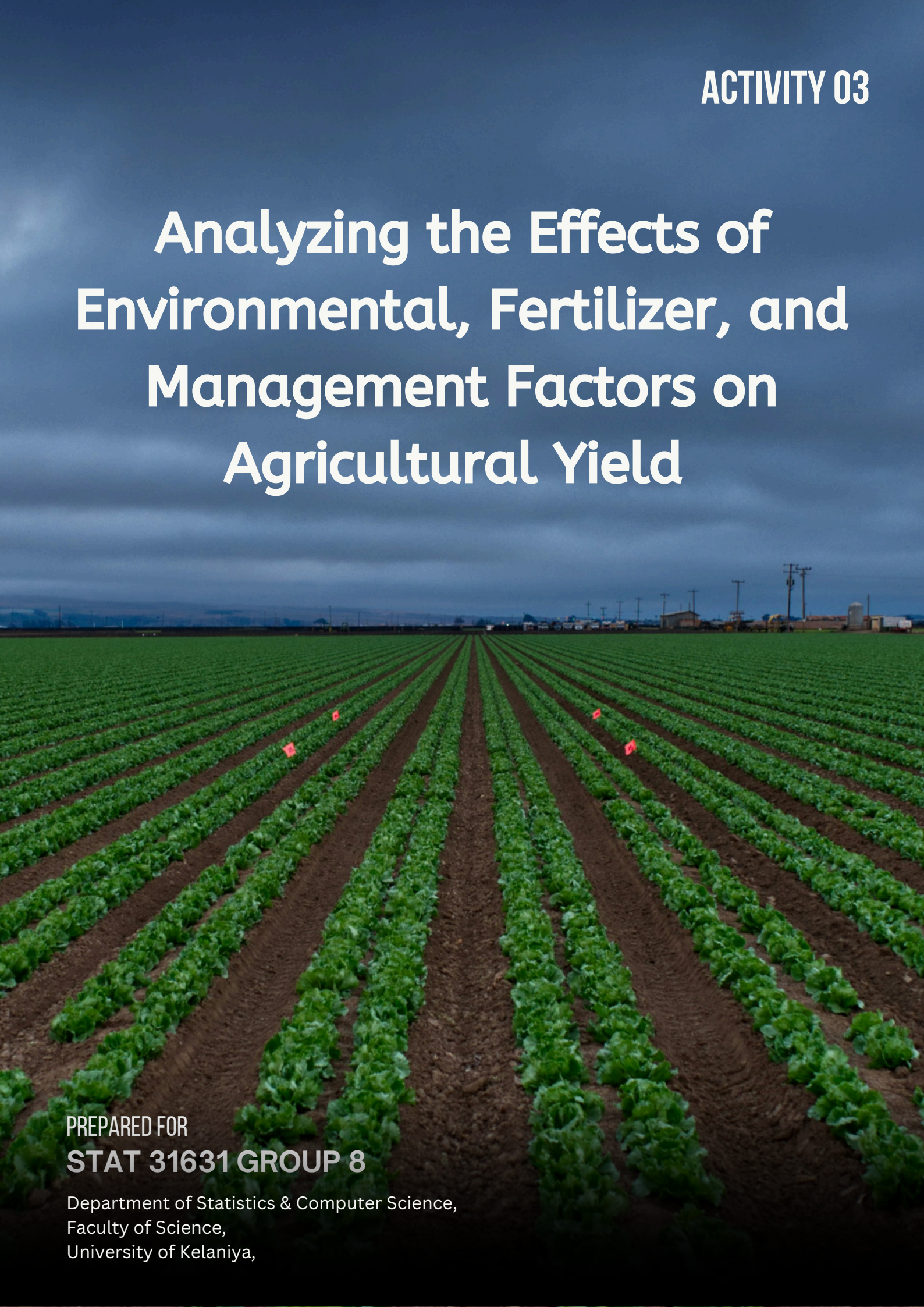
ACTIVITY 03

Analyzing the Effects of Environmental, Fertilizer, and Management Factors on Agricultural Yield



PREPARED FOR

STAT 31631 GROUP 8

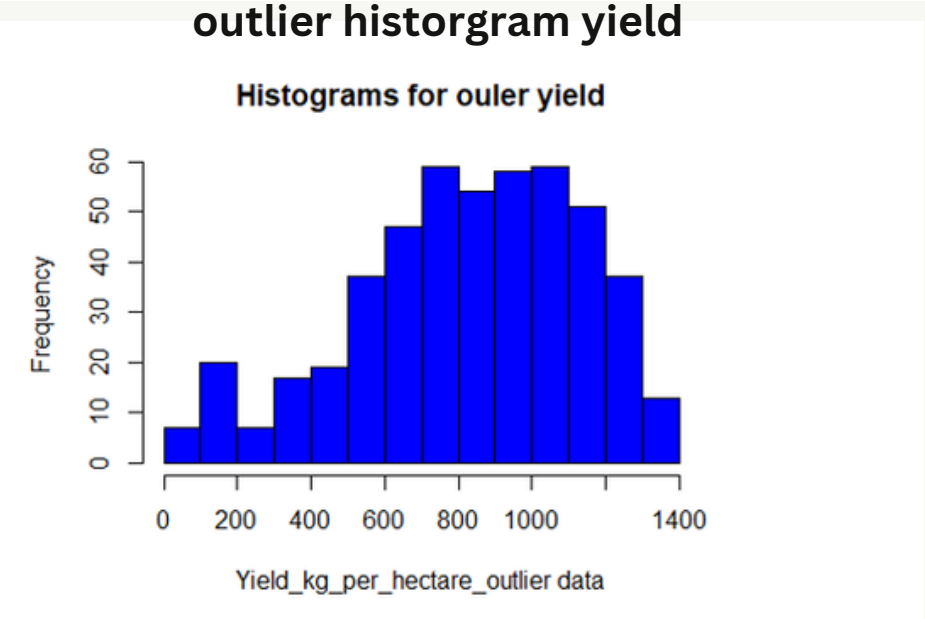
Department of Statistics & Computer Science,
Faculty of Science,
University of Kelaniya,

AGENDA

```
# Print the number of rows before and after removing outliers
print(paste("Number of rows before removing outliers:", nrow(data)))

## [1] "Number of rows before removing outliers: 16000"
```

```
print(paste("Number of rows after removing outliers:", nrow(cleaned_data)))

## [1] "Number of rows after removing outliers: 15515"

#consider the oulier of data frame
hist(outliers_data$Yield_kg_per_hectare ,xlab= "Yield_kg_per_hectare_outlier
data",main = "Histograms for ouler yield",breaks = 12,col = "blue")
```

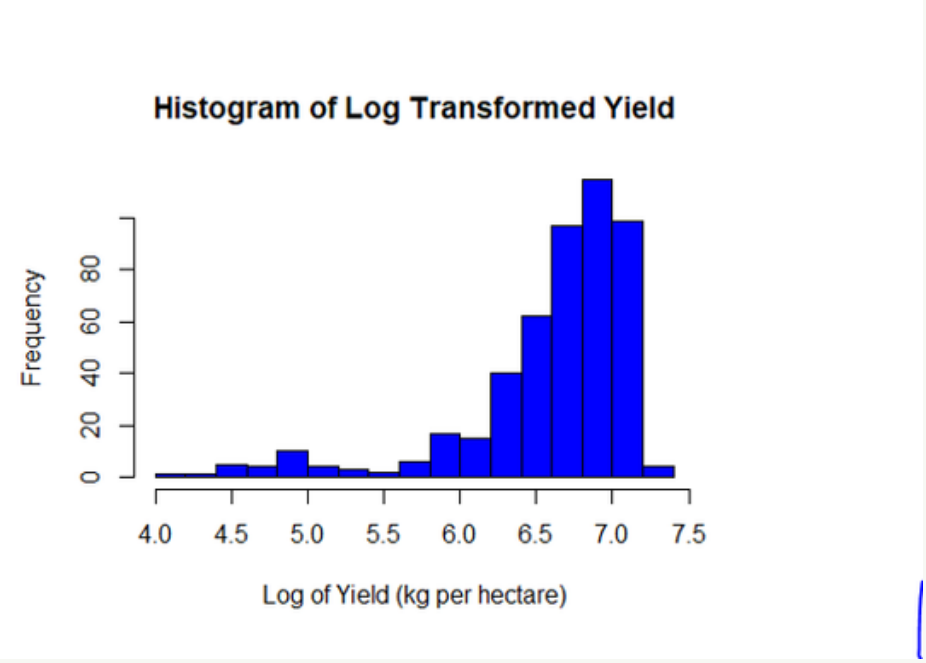
outlier historgram yield



The histogram is roughly bell-shaped but skewed slightly to the left (negatively skewed). This means most of the data points are concentrated in the middle to higher range of yield values, with fewer lower values. The histogram does include some very low yield values, which may be the outliers. The left skew indicates that a significant portion of the yields are higher, with fewer instances of very low yields.

outlier historgram (log tranfomasion)yield

```
# Logarithmic transformation
log_yield <- log(outliers_data$Yield_kg_per_hectare)
hist(log_yield, xlab = "Log of Yield (kg per hectare)", main = "Histogram of
Log Transformed Yield", breaks = 12, col = "blue")
```



The distribution has a clear peak around log values between 6.5 and 7.0. This histogram shows that after applying a logarithmic transformation to the yield data, the distribution is not more symmetric and not likely closer to normal.

```
#mean of the compair yeild
#with outlier mean of yeil
mean(data$Yield_kg_per_hectare)

## [1] 713.9997

#with out mean of yeil
mean(cleaned_data$Yield_kg_per_hectare)

## [1] 710.5186

#ouiltr mean
mean(outliers_data$Yield_kg_per_hectare)

## [1] 825.3593

#with oulier variyanse yeild
var(data$Yield_kg_per_hectare)

## [1] 40889.25

#with out oulier variyanse yeild
var(cleaned_data$Yield_kg_per_hectare)

## [1] 38814.39

#oulier variyanse yeild
var(outliers_data$Yield_kg_per_hectare)

## [1] 94665.92
```

Means: The outliers are higher than the general data points, as indicated by the mean of outliers being higher than the overall mean and the mean without outliers.

Variances: The presence of outliers significantly increases the variance, indicating that the outliers contribute to a greater spread in the data.

Removing the outliers leads to a more consistent dataset with a lower average yield and reduced variability. The outliers have a substantial effect on both the mean and variance, pushing both statistics higher.

hence we are consider the with out outlier data frame

data set divide in to two data frame train and test data frames

```
#consider the cleaned_data frame
# Calculate the number of samples for the training set (75% of the data)
train_size <- floor(0.75 * nrow(cleaned_data))

# Generate a vector of row indices
indices <- 1:nrow(cleaned_data)

# Randomly sample indices for the training set
train_indices <- sample(indices, size = train_size, replace = FALSE)

# Create training and testing sets
train1 <- cleaned_data[train_indices, ]
test1 <- cleaned_data[-train_indices, ]
```

```
# Print the number of rows train and testing data set
print(paste("Number of rows train data set:", nrow(train1)))

## [1] "Number of rows train data set: 11636"

print(paste("Number of rows teast data set:", nrow(test1)))

## [1] "Number of rows teast data set: 3879"
```

compair the seed varity

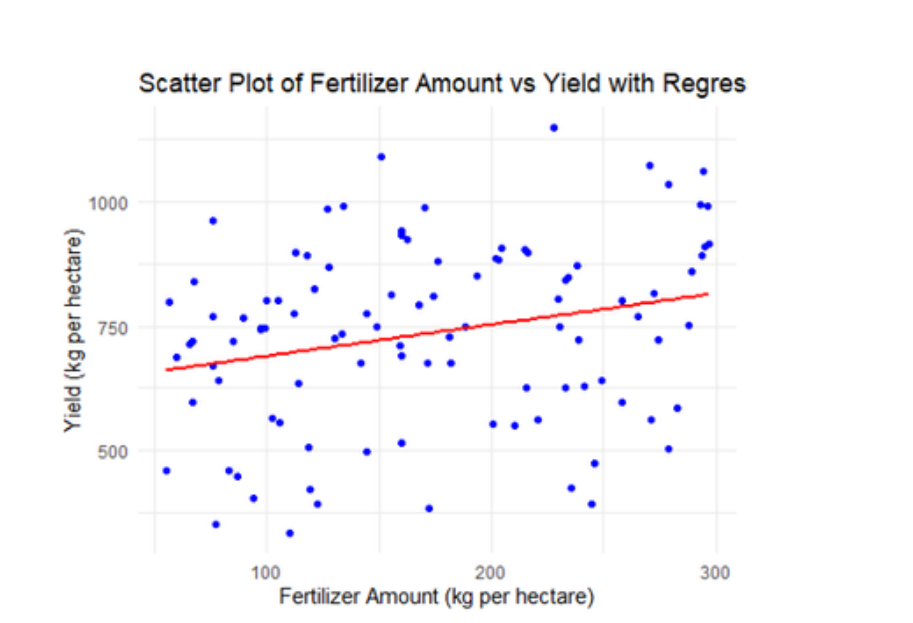
```
# Print the head of the filtered dataset to verify
head(train1_filtered_0)

##       Soil_Quality Seed_Variety Fertilizer_Amount_kg_per_hectare
Sunny_Days
## 15887     73.77213            0                          216.7756
100.36530
## 10203     55.29475            0                          205.9384
110.11012
## 3288      74.18859            0                          291.5795
77.83488
## 10039     70.83459            0                          298.1495
92.00413
## 4265      98.47427            0                          257.4484
93.08829
## 8489      72.22469            0                          122.0645
114.39440
##       Rainfall_mm Irrigation_Schedule Yield_kg_per_hectare
## 15887    434.1471                   2             373.2112
## 10203    432.5018                   7             626.8286
## 3288     521.5628                   4             472.3144
## 10039    560.1343                   3             474.9772
```

```
## 4265     366.0212                  3             519.3002
## 8489     351.2376                  3             490.3949

# Filter the train1 dataset where Seed_Variety is 1
train1_filtered_1 <- subset(train1, Seed_Variety == 1)

# Print the head of the filtered dataset to verify
head(train1_filtered_1)

##       Soil_Quality Seed_Variety Fertilizer_Amount_kg_per_hectare
Sunny_Days
## 8179      73.79380            1                         286.87839
78.39382
## 2685      83.96964            1                         208.89274
103.32797
## 5640      52.26928            1                          94.13869
105.75976
## 2945      52.95116            1                         264.06048
82.48267
## 6114      51.39387            1                         283.45747
113.54430
## 10929     65.27774            1                          91.05633
106.52755
##       Rainfall_mm Irrigation_Schedule Yield_kg_per_hectare
## 8179     510.9672                  7             903.6764
## 2685     653.4640                  3             705.3779
## 5640     746.5282                  5             522.3417
## 2945     466.4019                  8             996.3509
## 6114     453.8228                  7            1090.2711
## 10929    503.8265                  7             718.7664
```

our descriptive part include in activity 2

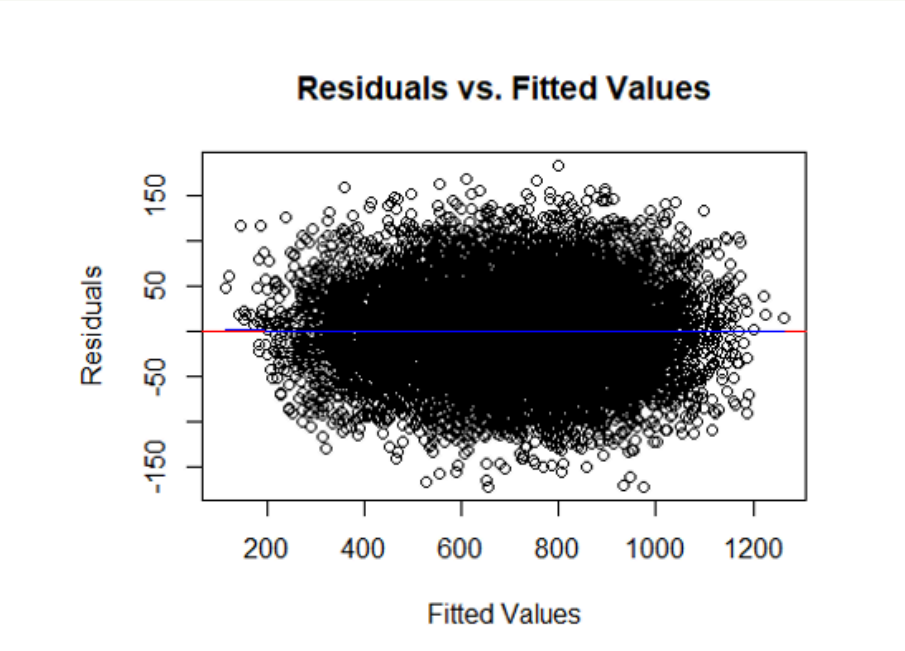
regression model creating part

```
# Sample 100 random rows from the cleaned_data data frame
set.seed(123) # Set seed for reproducibility
random_sample <- sample_n(cleaned_data, 100)

# Create scatter plot with a regression line for
Fertilizer_Amount_kg_per_hectare vs Yield_kg_per_hectare
ggplot(random_sample, aes(x = Fertilizer_Amount_kg_per_hectare, y =
Yield_kg_per_hectare)) +
  geom_point(color = "blue") + # Scatter plot
  geom_smooth(method = "lm", color = "red", se = FALSE) + # Regression line
  labs(title = "Scatter Plot of Fertilizer Amount vs Yield with Regression
Line",
       x = "Fertilizer Amount (kg per hectare)",
       y = "Yield (kg per hectare)") +
  theme_minimal()

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



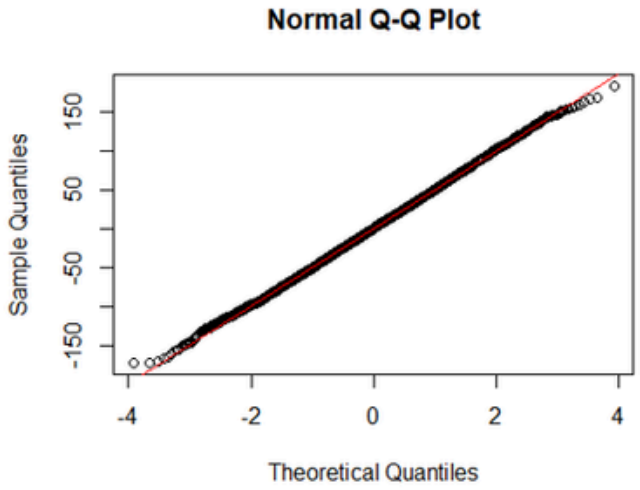
The red line represents the best-fit line through the data points. While the data points are scattered, there's a slight upward trend indicated by the regression line. This suggests a weakly positive correlation between fertilizer amount and yield.

```
# Fit the model
model <- lm(Yield_kg_per_hectare ~ Fertilizer_Amount_kg_per_hectare +
            Seed_Variety + Rainfall_mm + Irrigation_Schedule + Soil_Quality +
            Sunny_Days, data = train1)
#check the Diagnosing
#Diagnosing Heteroscedasticity
# Plot residuals vs. fitted values
#This plot helps detect non-linearity, unequal error variances
(heteroscedasticity), and outliers.
plot(model$fitted.values, model$residuals,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs. Fitted Values")
abline(h = 0, col = "red")

# Add a smoothed line to identify patterns
lines(lowess(model$fitted.values, model$residuals), col = "blue")
```



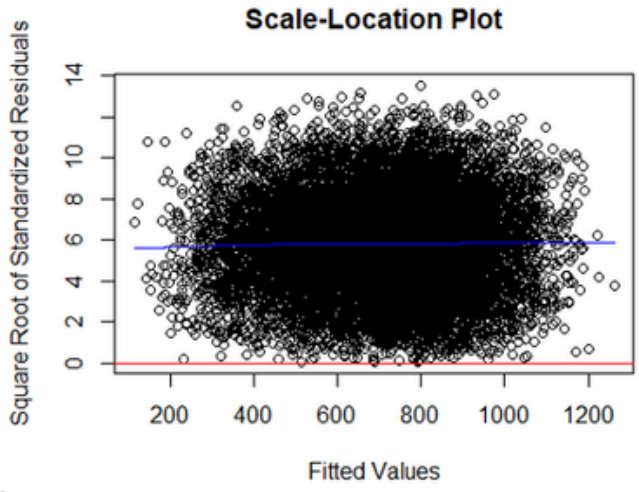
The residuals are randomly distributed. There are no clear patterns, curves. This indicates that the linear relationship between the variables is appropriate and there is no evidence of non-linearity. The spread of the points is relatively constant suggests that the variance of the errors is constant.

```
#Normal Q-Q Plot
qqnorm(model$residuals, main = "Normal Q-Q Plot")
qqline(model$residuals, col = "red")
```



The points on the plot mostly follow the red line .This plot indicates that the residuals of model are approximately normally distributed, with a few outliers present.

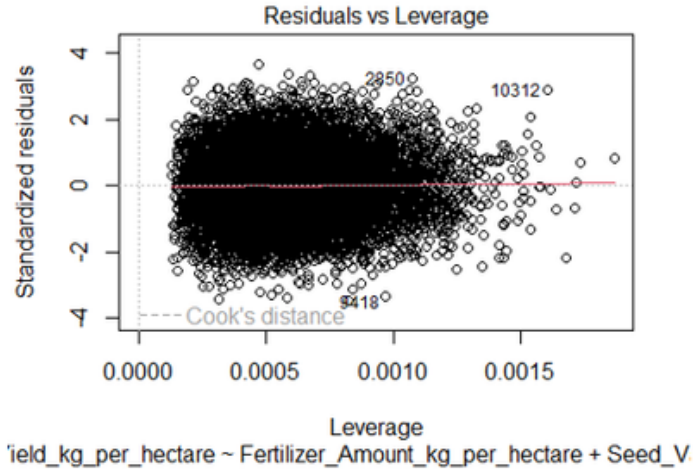
```r
#Scale-Location Plot (Spread-Location Plot)
plot(model$fitted.values, sqrt(abs(model$residuals)),
     xlab = "Fitted Values",
     ylab = "Square Root of Standardized Residuals",
     main = "Scale-Location Plot")
abline(h = 0, col = "red")
lines(lowess(model$fitted.values, sqrt(abs(model$residuals))), col = "blue")
```



The points representing the square root of standardized residuals are scattered randomly around the red line without any clear patterns. This further supports the conclusion of constant variance. the scale-location plot suggests that the linear regression model is appropriate for the data and that the constant variance assumption holds

The points are randomly scattered without any patterns. There are a few points with relatively large standardized residuals.This plot suggests that the model is reasonably well-behaved.

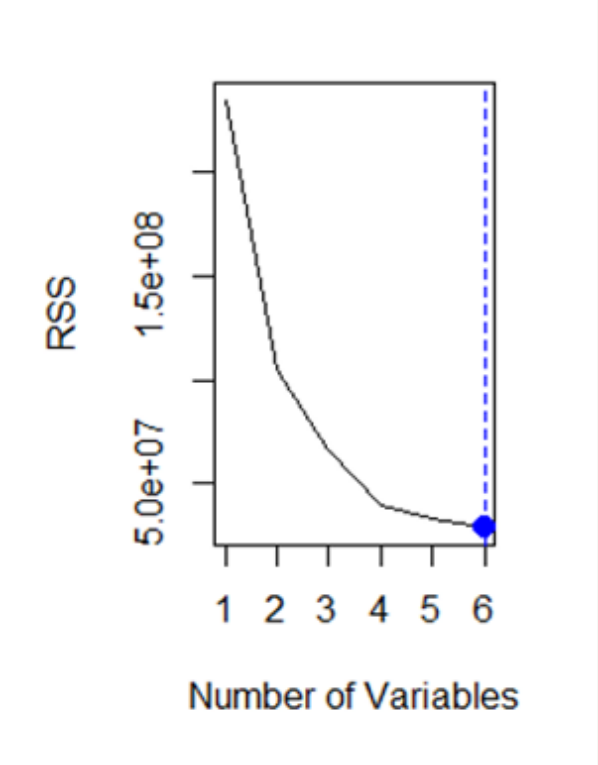
```r
#Residuals vs. Leverage Plot

plot(model, which = 5)
```



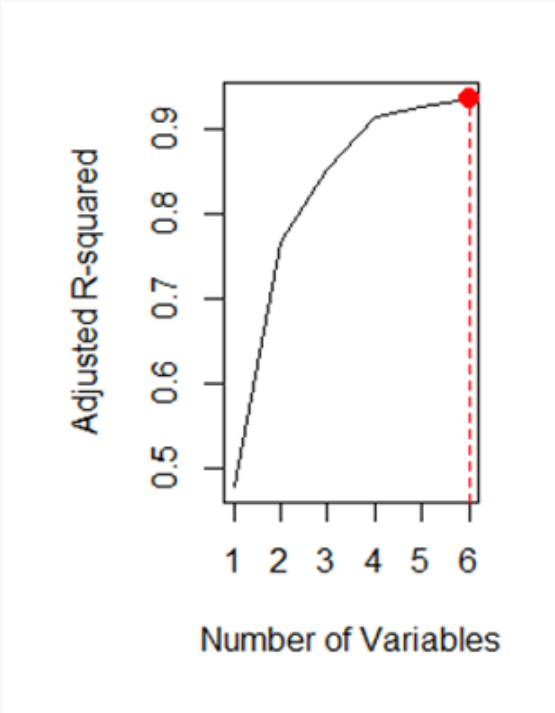
```r
#forwer selcsion method
library(leaps)

## Warning: package 'leaps' was built under R version 4.4.1

# Fit the regsubsets model with all predictors including Seed_Variety
Model_forward<- regsubsets(Yield_kg_per_hectare ~
Fertilizer_Amount_kg_per_hectare + Seed_Variety+ Rainfall_mm +
Irrigation_Schedule + Soil_Quality+Sunny_Days, data = train1,nvmax =
6,method="forward")  # nvmax is the maximum number of predictors to include

# Get the summary of the model
model_with_seed_summary <- summary(Model_forward)

# Display key information from the summary
print(model_with_seed_summary)

## Subset selection object
## Call: regsubsets.formula(Yield_kg_per_hectare ~
Fertilizer_Amount_kg_per_hectare +
##     Seed_Variety + Rainfall_mm + Irrigation_Schedule + Soil_Quality +
##     Sunny_Days, data = train1, nvmax = 6, method = "forward")
## 6 Variables  (and intercept)
```

```r
= 2, pch = 20)
abline(v = adjr2_max, col = "red", lty = 2)
```

The RSS generally decreases as the number of variables increases. This is expected because adding more variables to a model typically explains more of the variation in the data. The vertical line with a blue dot likely represents the chosen model based on this plot.

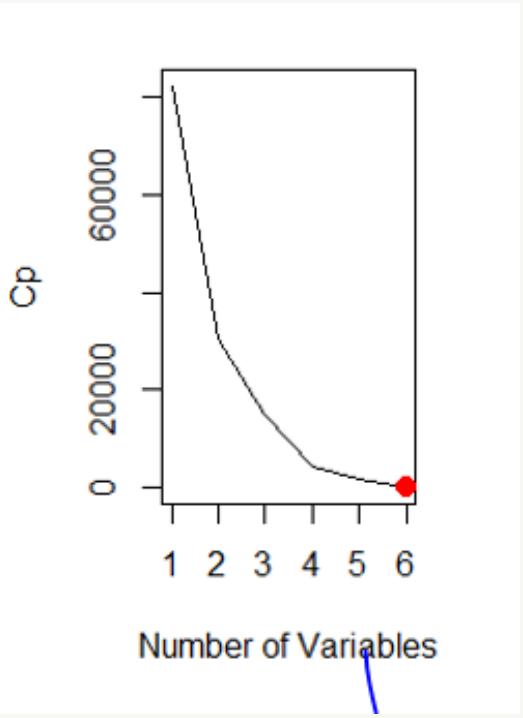


Adjusted R-squared generally increases with the number of variables, but after a certain point, it starts to decrease due to overfitting. The vertical line with a red dot likely represents the chosen model based on this plot.

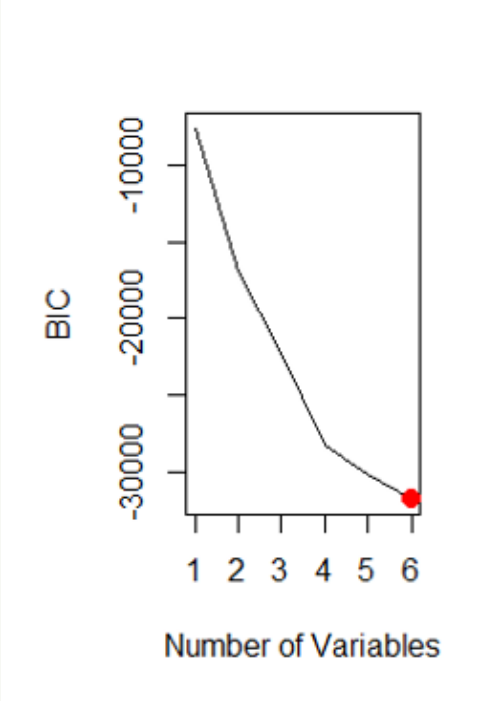



```
par(mfrow = c(1, 2))
plot(model_with_seed_summary$cp, xlab = "Number of Variables", ylab = "Cp",
type = "l")
cp.min <- which.min(model_with_seed_summary$cp)
points(cp.min, model_with_seed_summary$cp[cp.min], col = "red", cex = 2, pch
= 20)
bic.min <- which.min(model_with_seed_summary$bic)
plot(model_with_seed_summary$bic, xlab = "Number of Variables", ylab = "BIC",
type = "l")
points(bic.min, model_with_seed_summary$bic[bic.min], col = "red", cex = 2,
pch = 20)
```

As the number of variables increases, the Cp value initially decreases rapidly. This suggests that adding more variables significantly improves the model's fit. The red dot on the graph likely indicates the chosen model based on the Cp criterion



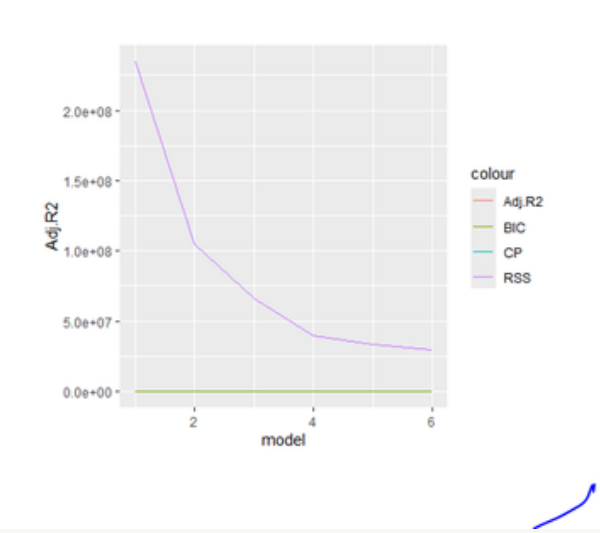
The BIC value decreases rapidly as the number of variables increases initially, suggesting that adding more variables improves model fit. The red dot marks the model with the lowest BIC value, which is often considered the optimal model



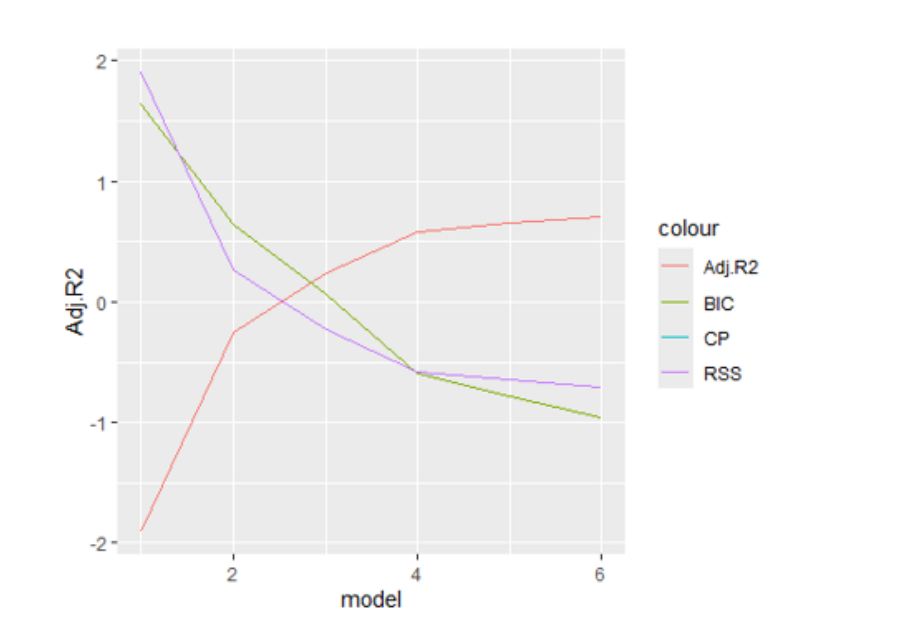

```
res.sum <- summary(Model_forward)
criterion<-data.frame(
model=1:6,
Adj.R2 = (res.sum$adjr2),
CP = (res.sum$cp),
BIC = (res.sum$bic),
RSS=res.sum$rss
)
head(criterion)
```

```
##   model    Adj.R2        CP        BIC       RSS
## 1     1 0.4783875 82318.723  -7555.337 234753559
## 2     2 0.7667792 30373.167 -16913.251 104952816
## 3     3 0.8528503 14871.444 -22263.678  66213847
## 4     4 0.9127561  4083.971 -28337.945  39254317
## 5     5 0.9255808  1775.468 -30179.645  33481091
## 6     6 0.9354083     7.000 -31819.256  29057245
```

```
library(ggplot2)
ggplot(criterion, aes(model)) +
 geom_line(aes(y = Adj.R2, colour = "Adj.R2")) +
 geom_line(aes(y = CP, colour = "CP"))+
 geom_line(aes(y = BIC, colour = "BIC"))+
 geom_line(aes(y = RSS, colour = "RSS"))
```



```
# standardize
criterion_std<-cbind(model=criterion$model, scale(criterion[,-1]))
criterion_std<-as.data.frame(criterion_std)
head(criterion_std)

##   model     Adj.R2         CP        BIC        RSS
## 1     1 -1.9050862  1.9051292  1.6414660  1.9050931
## 2     2 -0.2580365  0.2579539  0.6368103  0.2580227
## 3     3  0.2335289 -0.2336003  0.0623942 -0.2335447
## 4     4  0.5756598 -0.5756672 -0.5897326 -0.5756401
## 5     5  0.6489041 -0.6488691 -0.7874555 -0.6488979
## 6     6  0.7050300 -0.7049465 -0.9634824 -0.7050330

#after stadlize
ggplot(criterion_std, aes(model)) +
 geom_line(aes(y = Adj.R2, colour = "Adj.R2")) +
 geom_line(aes(y = CP, colour = "CP"))+
 geom_line(aes(y = BIC, colour = "BIC"))+
 geom_line(aes(y = RSS, colour = "RSS"))
```



We would ideally choose a model with a relatively high Adjusted R-squared, while keeping BIC, CP, and RSS at reasonably low levels.

```
coef(Model_forward, 6)

##                      (Intercept) Fertilizer_Amount_kg_per_hectare
##                       44.0639085                         0.8076244
##                     Seed_Variety                       Rainfall_mm
##                      300.2585927                        -0.5032444
##              Irrigation_Schedule                      Soil_Quality
##                       49.7783048                         1.5537554
##                       Sunny_Days
##                        2.0301833

#better model
better_model <- lm(Yield_kg_per_hectare ~ Fertilizer_Amount_kg_per_hectare +
Seed_Variety+ Rainfall_mm + Irrigation_Schedule +
Soil_Quality+Sunny_Days,data=train1 )
summary(better_model)

##
## Call:
## lm(formula = Yield_kg_per_hectare ~ Fertilizer_Amount_kg_per_hectare +
##     Seed_Variety + Rainfall_mm + Irrigation_Schedule + Soil_Quality +
##     Sunny_Days, data = train1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -172.752  -33.980    0.045   33.440  183.162
##
```

```
## Coefficients:
##                                   Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                      44.063908   6.160398    7.153 9.01e-13
***
## Fertilizer_Amount_kg_per_hectare  0.807624   0.006455  125.109  < 2e-16
***
## Seed_Variety                    300.258593   1.017991  294.952  < 2e-16
***
## Rainfall_mm                      -0.503244   0.004809 -104.651  < 2e-16
***
## Irrigation_Schedule              49.778305   0.220689  225.559  < 2e-16
***
## Soil_Quality                      1.553755   0.032012   48.537  < 2e-16
***
## Sunny_Days                        2.030183   0.048249   42.077  < 2e-16
***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.99 on 11629 degrees of freedom
## Multiple R-squared:  0.9354, Adjusted R-squared:  0.9354
## F-statistic: 2.808e+04 on 6 and 11629 DF,  p-value: < 2.2e-16
```

Fitted model

Yield_kg_per_hectare = 44.063908 + 0.807624(Fertilizer_Amount_kg_per_hectare) + 300.258593 (Seed_Variety) - 0.503244 (Rainfall_mm) + 49.778305 (Irrigation_Schedule) + 1.553755(Soil_Quality) + 2.030183 (Sunny_Days)

Fertilizer Amount (β=0.8076):
The coefficient for fertilizer amount is positive and highly significant (p<2e−16p < 2e-16p<2e−16). For every additional kilogram of fertilizer per hectare, the yield is expected to increase by approximately 0.81 kg per hectare, holding other factors constant.

Seed Variety (β=300.26):
The seed variety variable also shows a highly significant positive effect (p<2e−16p < 2e-16p<2e−16) on yield.

Irrigation Schedule (β=49.78):
The positive and highly significant coefficient for irrigation schedule (p<2e−16p < 2e-16p<2e−16) .

Sunny Days (β=2.0302):
The number of sunny days also positively affects yield, with a significant coefficient (p<2e−16p < 2e-16p<2e−16).

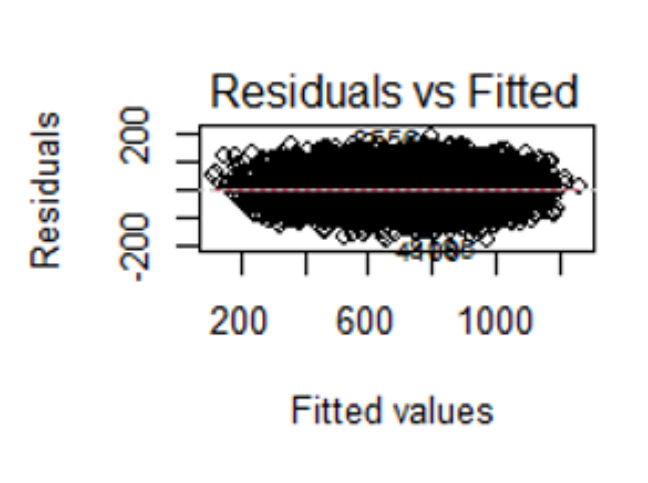
The model's adjusted R2R^2R2 value was 0.9354, indicating that approximately 93.5% of the variability in agricultural yield could be explained by these factors. This high R2R^2R2 value suggests a strong relationship between the predictors and yield, making the model a valuable tool for understanding and predicting crop productivity.

The residual standard error of 49.99 further supports the model's accuracy, indicating that the predictions are reasonably close to the observed values.

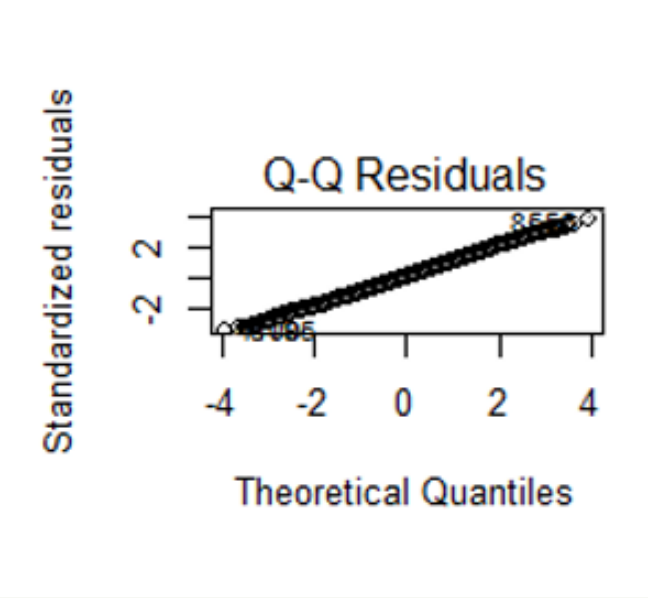
The F-statistic (28,080.00) with a p-value of < 2.2e-16 indicates that the model is highly significant.

Residual Analysis

```
# Model diagnostics
par(mfrow=c(2,2))
plot(better_model)
```

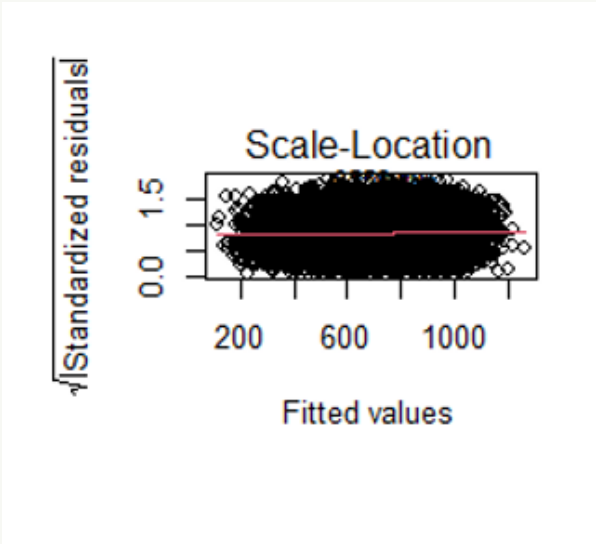


The points are generally scattered randomly. The spread of the points seems relatively constant. There are a few points that deviate significantly from the main cluster. These points could be outliers. The plot suggests that the model assumptions are reasonably met, with the exception of outliers.

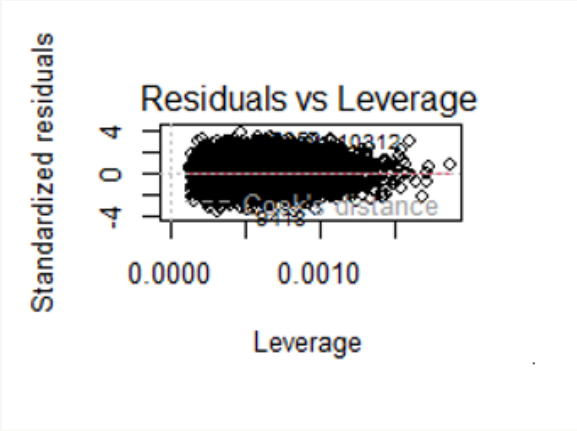


The points deviate from a straight line, especially in the tails. This suggests that the residuals are not normally distributed.

RESULTS AND DISCUSSON



The points are generally scattered randomly without any clear pattern. The scale-location plot indicates that the assumption of homoscedasticity is likely satisfied in this model. The variance of the residuals appears to be constant across different levels of the fitted values.



The points are scattered randomly without any pattern. the plot suggests that there are no major concerns about influential points or heteroscedasticity in the model.

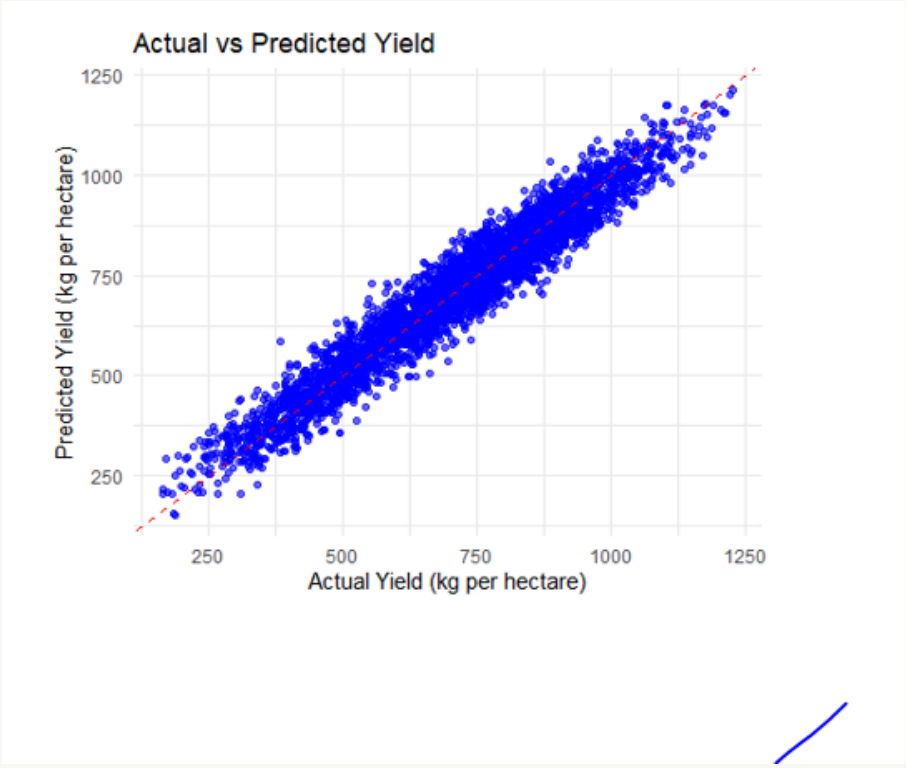
```
#####################################################################

# Make predictions on the test data
test_predictions <- predict(better_model, newdata = test1)



#mean yeil of the focat yeil
mean(test_predictions)

## [1] 711.9286

#mean of the actual yeild
mean(test1$Yield_kg_per_hectare)

## [1] 712.5346

# Calculate performance metrics
mae <- mean(abs(test_predictions - test1$Yield_kg_per_hectare))
rmse <- sqrt(mean((test_predictions - test1$Yield_kg_per_hectare)^2))
r_squared <- cor(test_predictions, test1$Yield_kg_per_hectare)^2

cat("Test MAE:", mae, "\nTest RMSE:", rmse, "\nTest R-squared:", r_squared,
"\n")

## Test MAE: 39.78421
## Test RMSE: 49.59865
## Test R-squared: 0.9372522

#focat_value add teast one data set

test1$focat_yeild=test_predictions
head(test1)
```

```
##    Soil_Quality Seed_Variety Fertilizer_Amount_kg_per_hectare Sunny_Days
## 8      69.33589            1                         135.92277  119.82700
## 16     77.14188            0                         286.16030   89.06296
## 17     71.57169            0                          91.82663  101.39229
## 19     78.76663            1                         239.54935  101.53371
## 25     83.85147            1                         133.72629   97.19609
## 26     53.63000            1                         298.46915  100.00733
##    Rainfall_mm Irrigation_Schedule Yield_kg_per_hectare focat_yeild
## 8     384.3504                   2             750.3530    711.2333
## 16    650.7772                   5             436.2702    497.2393
## 17    494.6965                   5             372.0941    435.2135
## 19    429.3474                   3             698.1650    799.5728
## 25    547.8296                   6             877.4256    802.9113
## 26    489.8564                   9            1069.6259   1073.2219
```

When applying the model to the test dataset, the Mean Absolute Error (MAE) was 39.78 kg per hectare, and the Root Mean Squared Error (RMSE) was 49.60 kg per hectare. These metrics suggest that the model predictions are generally close to the actual yield values. The adjusted R^2 value on the test data was 0.9373, indicating that the model's predictive power remained strong even on unseen data

```
# Plot actual vs. predicted values
ggplot(test1, aes(x = test1$Yield_kg_per_hectare, y = test1$focat_yeild)) +
  geom_point(color = "blue", alpha = 0.6) +
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") +
  labs(title = "Actual vs Predicted Yield",
       x = "Actual Yield (kg per hectare)",
       y = "Predicted Yield (kg per hectare)") +
  theme_minimal()
```

```
## Warning: Use of `test1$Yield_kg_per_hectare` is discouraged.
## ℹ Use `Yield_kg_per_hectare` instead.

## Warning: Use of `test1$focat_yeild` is discouraged.
## ℹ Use `focat_yeild` instead.
```



This plot which is comparing actual versus predicted yields showed that the predictions closely aligned with the observed values, further validating the model's accuracy.

The summary statistics of the predicted yields were similar to those of the actual yields, with mean values of 711.9 kg per hectare for the predictions and 712.5 kg per hectare for the actual yields.

```
#get sumriy actual vs. predicted values
#actua
summary(test1$Yield_kg_per_hectare)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   164.2   571.9   734.2   712.5   858.9  1229.0

#predicted values
summary(test1$focat_yeild)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   151.8   582.5   734.3   711.9   853.3  1212.8
```

THE TEAM

B.L.L. OSHAN	PS/2020/023
W.K.H.RUKSHANI	PS/2020/316
B.D.H.CHATHURANGA	PS/2020/141
N.D.K.NADEESHA	PS/2020/258
W.K.S.LAKMALI	PS/2020/186
A.S.S.SILVA	PS/2020/185
T.M.S.D.THENNAKOON	PS/2020/306
P.S.A.LIYANAGE	PS/2020/260