

# Infertility After Abortion: Insights from Data Analysis in



Data Insights and Analysis by

**Muhammad Najeeb Haider**

Date: 08-10-2024



[Nexa Tunnel](#)



[Najeeb Haider](#)



"This work is dedicated to those who dream of holding a child but have faced the painful reality of infertility. To everyone who has endured the sorrow of miscarriage or abortion, your strength and perseverance inspire this project. May it shine a light on the challenges you face and honor your hope, strength, and unyielding spirit."



Infertility After Abortion: Insights from Data Analysis in .....	1
Introduction: .....	5
List of libraries: .....	7
Data Preprocessing: .....	8
Exploratory Data Analysis (EDA) .....	9
• Frequency of categorical variables.....	9
• Summary: .....	10
• Library survival .....	10
• Chi-square Test.....	10
• Effect of Age and Education on Infertility .....	11
Bar Plot: Distribution of Infertility Cases (case) .....	11
Histogram: Distribution of Age: .....	12
Boxplot: Age vs. Infertility (case):.....	13
Bar Plot: Parity (Number of Children) by Infertility Case: .....	13
Scatter Plot: Age vs Parity, Colored by Infertility (case) .....	14
Faceted Bar Plot: Induced and Spontaneous Abortions by Infertility .....	15
Pair Plot (Relationships Between Multiple Variables) .....	16
Advance Analysis: .....	18
Build a Decision Tree:.....	18
Build a Neural Network:.....	21
Logistic Regression .....	23
The k-nearest neighbors (KNN) algorithm:.....	31
XGBoost (Extreme Gradient Boosting) .....	38
Confusion Matrix Heatmap: .....	39
Model Evaluation .....	40
Insights from the Data on "Infertility after Spontaneous and Induced Abortion" .....	41
Key Insights and Interpretations .....	44
Conclusion.....	44

Explaining the Challenge in data Stable Modeling Infertility After Spontaneous and Induced Abortion .....	45
From Frustration to Fulfillment: My Overall Experience with Data.....	47

## Introduction

### *Objective of the Project*

This project, "Infertility after Spontaneous and Induced Abortion," is designed to analyze the potential link between different types of abortion and subsequent infertility. I chose to conduct the analysis in **R** to showcase my proficiency in this powerful statistical programming language, which is widely used in the field of **Data Science**. R's versatility in data manipulation, statistical analysis, and visualization makes it an essential tool for analyzing complex healthcare datasets. By demonstrating my skills in R, I aim to highlight my ability to handle real-world datasets, conduct thorough statistical analyses, and create meaningful insights that can drive informed decision-making in my **Data Science** career.

### *Problem Statement*

- Infertility following spontaneous or induced abortion is a critical public health issue that affects millions of women worldwide. Understanding the potential risk factors and long-term effects of abortion on fertility is essential for improving healthcare interventions. This study focuses on identifying any significant links between abortion histories and infertility, which could provide actionable insights for healthcare professionals to offer better counseling, support, and treatment for women at risk. Addressing this issue not only has the potential to improve individual health outcomes but also contributes to a broader understanding of reproductive health in the population.

*Key Focus:* Blend of data storytelling and technical expertise in data analysis and visualization.

## Data Sources & Methodology

- **Data Collection:**

This dataset has been taken from the R database, where it is available under the name "Infert." It primarily consists of observations of 248 individuals from the United States who have experienced miscarriage, either induced or spontaneous, at some point in their lives.

**Data()**

Data sample "infert".

	education	age	parity	induced	case	spontaneous	stratum	pooled.stratum
1	0-5yrs	26	6	1	1	2	1	3
2	0-5yrs	42	1	1	1	0	2	1
3	0-5yrs	39	6	2	1	0	3	4
4	0-5yrs	34	4	2	1	0	4	2
5	6-11yrs	35	3	1	1	1	5	32
6	6-11yrs	36	4	2	1	1	6	36
7	6-11yrs	23	1	0	1	0	7	6
8	6-11yrs	32	2	0	1	0	8	22
9	6-11yrs	21	1	0	1	1	9	5
10	6-11yrs	28	2	0	1	0	10	19
11	6-11yrs	29	2	1	1	0	11	20
12	6-11yrs	37	4	2	1	1	12	37
13	6-11yrs	31	1	1	1	0	13	9
14	6-11yrs	29	3	2	1	0	14	29
15	6-11yrs	31	2	1	1	1	15	21
16	6-11yrs	27	2	2	1	0	16	18
17	6-11yrs	30	5	2	1	1	17	38
18	6-11yrs	26	1	0	1	1	18	7
19	6-11yrs	25	3	2	1	1	19	28
20	6-11yrs	44	1	0	1	1	20	17
21	6-11yrs	40	1	0	1	1	21	14

### Column Breakdown:

1. **education:** This represents the education level of the participants (e.g., years of education).
2. **age:** The age of the participants at the time of data collection.
3. **parity:** The number of times a woman has given birth to a viable child (full-term pregnancies).
4. **induced:** A binary or categorical variable indicating whether the participant had an induced abortion (number of induced abortions).
5. **case:** Represent whether the participant experienced infertility after the abortion (1 = case of infertility, 0 = no infertility).
6. **spontaneous:** This refers to spontaneous abortions (miscarriages). Similar to induced, indicating whether the participant had experienced a spontaneous abortion.

7. **stratum:** Represents a grouping variable used to categorize participants into different strata, often for matching or controlling variables such as age groups or socioeconomic status.
8. **pooled. Stratum:** Similar to stratum but involve combining multiple groups or categories for analysis purposes, particularly in stratified or pooled data analysis.

- **Data Cleaning:**

This data was already clean and pre-prepared. But I check it I don't think there were any missing values in it. Summarize the steps we took to clean and prepare the data for analysis.

I named this data `I <- infert` then

```
>
> # Check for missing values
> sum(is.na(I))
[1] 0
> |
```

- **Tools Used:**

The tools used in this dataset primarily involved the R language. The tools that were used include those that were first installed, and then, whenever a tool was needed, it was call into the library for use.

### List of libraries:

- survival ggplot2 ,dplyr, tidyr ,purrr, gridExtra, cowplot,
  - MASS, ggcorrplot, GGally, skimr, dplyr,
  - rpart, rpart.plot, nnet, caret, NeuralNetTools, randomForest, e1071, xgboost
- In this list some libraries were used for data potential analysis some are used for advance tasks like Neural Network and machine learning tasks. Some of these libraries was used in several times.

## Data Preprocessing:

Identifying data structure, class of data variables

Convert categorical variables (e.g., education, induced) to numeric format using encoding techniques like label encoding.

```
> str(I)
'data.frame': 248 obs. of 8 variables:
 $ education   : Factor w/ 3 levels "0-5yrs","6-11yrs",...: 1 1 1 1 2 2 2 2 2 ...
 $ age         : num  26 42 39 34 35 36 23 32 21 28 ...
 $ parity      : num  6 1 6 4 3 4 1 2 1 2 ...
 $ induced     : num  1 1 2 2 1 2 0 0 0 0 ...
 $ case        : num  1 1 1 1 1 1 1 1 1 1 ...
 $ spontaneous : num  2 0 0 0 1 1 0 0 1 0 ...
 $ stratum     : int   1 2 3 4 5 6 7 8 9 10 ...
 $ pooled.stratum: num   3 1 4 2 32 36 6 22 5 19 ...

> |

> sapply(I, class)
      education      age      parity      induced      case  spontaneous      stratum
      "factor"    "numeric"  "numeric"  "numeric"  "numeric"  "numeric"  "integer"
pooled.stratum
      "numeric"
> I[1:5,]
  education age parity induced case spontaneous stratum pooled.stratum
1  0-5yrs  26     6     1     1     2         1         3
2  0-5yrs  42     1     1     1     0         2         1
3  0-5yrs  39     6     2     1     0         3         4
4  0-5yrs  34     4     2     1     0         4         2
5  6-11yrs 35     3     1     1     1         5        32
```

After some changing:

```
I$education <- as.factor(I$education)
```

```
I$induced <- as.factor(I$induced)
```

```
# Apply label encoding using caret's preProcess function
```

```
preproc <- preProcess(I, method = c("center", "scale"))
```

```
I_numeric <- predict(preproc, I)
```



```

> I[5,]
  education age parity induced case spontaneous stratum
5   6-11yrs  35     3      1      1           1       5
  pooled.stratum predicted_probs
5              32      0.4993621
> I[1:5,]
  education age parity induced case spontaneous stratum
1   0-5yrs  26     6      1      1           2       1
2   0-5yrs  42     1      1      1           0       2
3   0-5yrs  39     6      2      1           0       3
4   0-5yrs  34     4      2      1           0       4
5   6-11yrs  35     3      1      1           1       5
  pooled.stratum predicted_probs
1              3      0.33574094
2              1      0.46556392
3              4      0.06586586
4              2      0.18238904
5             32      0.49936211
> |

```

## Exploratory Data Analysis (EDA)

**Objective:** Understand the distribution of the data and relationships between variables.

- Cross-tabulations:

```
table(I$case, I$induced)
```

```
table(I$case, I$spontaneous)
```

```

> # Cross-Tabulations:
> table(I$case, I$induced)

      0  1  2
0  96 45 24
1  47 23 13
> table(I$case, I$spontaneous)

      0    1    2
0  113   40   12
1   28   31   24
> |

```

- Frequency of categorical variables

```
table(I$case)
```

```
table(I$induced)
```

```
table(I$spontaneous)
```

Due to dull screenshot, I make values in table form to show you.

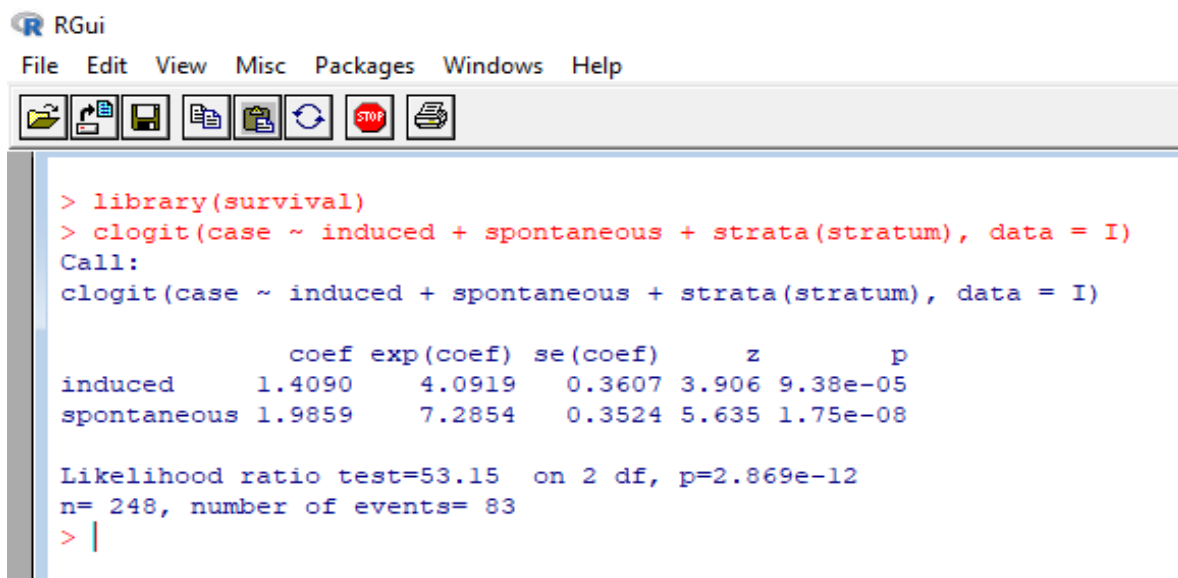
table(I\$case)		table(I\$induced)			table(I\$spontaneous)		
0	1	0	1	2	0	1	2
165	83	143	68	37	141	71	36

- Summary:

```
> summary(I)
  education      age      parity induced      case      spontaneous      stratum
0-5yrs : 12   Min.   :21.00   Min.   :1.000  0:143   Min.   :0.0000   Min.   :0.0000   Min.   : 1.00
6-11yrs:120  1st Qu.:28.00   1st Qu.:1.000  1: 68   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:21.00
12+ yrs:116  Median :31.00   Median :2.000  2: 37   Median :0.0000   Median :0.0000   Median :42.00
              Mean   :31.50   Mean   :2.093           Mean   :0.3347   Mean   :0.5766   Mean   :41.87
              3rd Qu.:35.25   3rd Qu.:3.000           3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:62.25
              Max.   :44.00   Max.   :6.000           Max.   :1.0000   Max.   :2.0000   Max.   :83.00

pooled.stratum
Min.   : 1.00
1st Qu.:19.00
Median :36.00
Mean   :33.58
3rd Qu.:48.25
Max.   :63.00
> |
```

- Library survival



```
> library(survival)
> clogit(case ~ induced + spontaneous + strata(stratum), data = I)
Call:
clogit(case ~ induced + spontaneous + strata(stratum), data = I)

              coef exp(coef) se(coef)      z      p
induced      1.4090    4.0919  0.3607  3.906 9.38e-05
spontaneous  1.9859    7.2854  0.3524  5.635 1.75e-08

Likelihood ratio test=53.15 on 2 df, p=2.869e-12
n= 248, number of events= 83
> |
```

- Chi-square Test:

```
chisq.test(table(I$induced, I$case))
```

Pearson's Chi-squared test

data: table(I\$induced, I\$case)

X-squared = 0.07323, df = 2, p-value = 0.964

- Effect of Age and Education on Infertility

`aggregate(case ~ age, data = I, mean)`

Here are some samples:

	Age	Case
1	21	0.3333333
2	23	0.3333333
3	24	0.3333333
4	30	0.3333333
5	38	0.3750000
6	42	0.3333333
7	44	0.3333333

`aggregate(case ~ education, data = I, mean)`

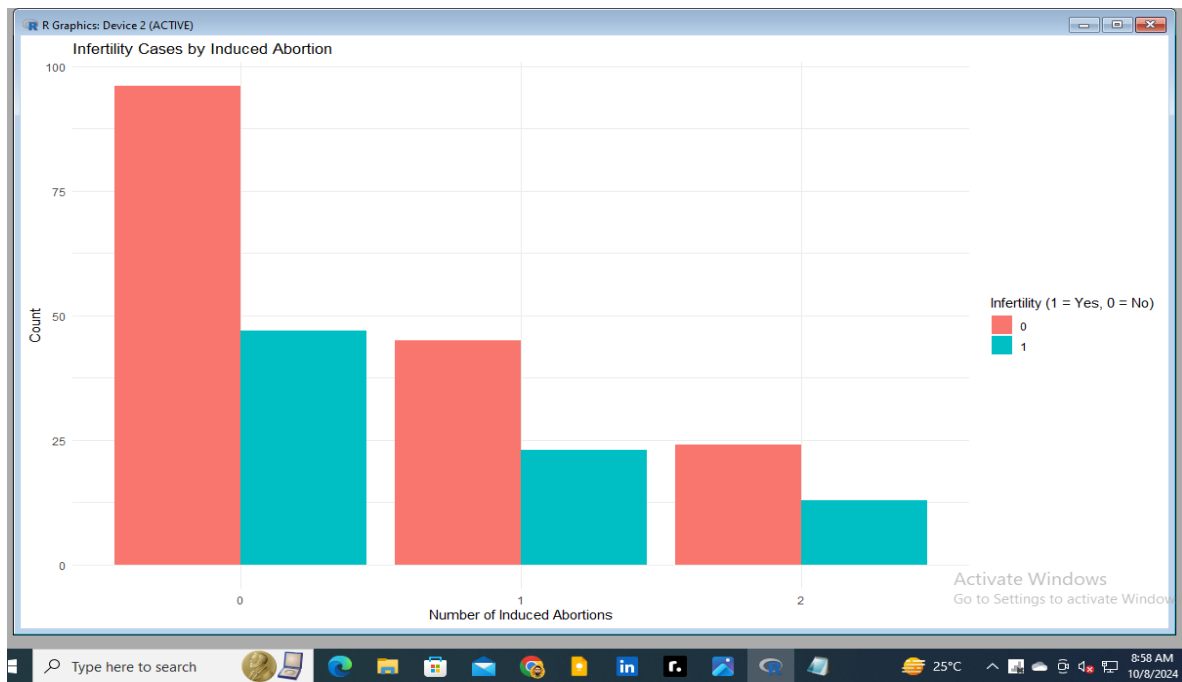
	Education	Case
1	o- 5yrs	0.3333333
2	2 6-11yrs	0.3333333
3	12+ yrs	0.3362069

## Visualizations:

### Bar Plot: Distribution of Infertility Cases (case)

Visualize the count of women who experienced infertility (case variable) based on whether they had an induced abortion (induced).

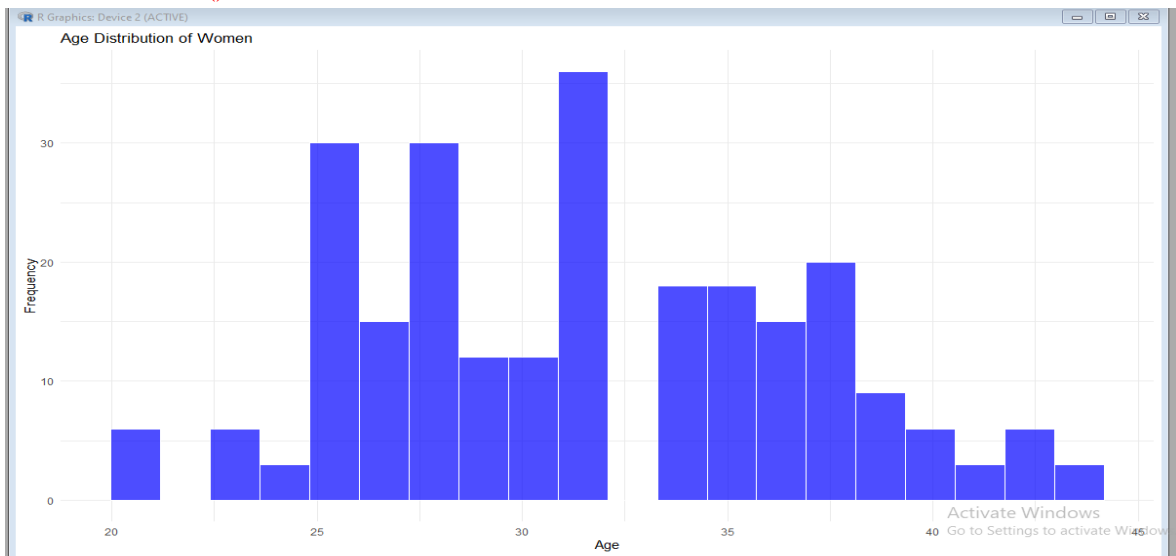
- `ggplot(I, aes(x = factor(induced), fill = factor(case))) +  
geom_bar(position = "dodge") +  
labs(title = "Infertility Cases by Induced Abortion",  
x = "Number of Induced Abortions",  
y = "Count",  
fill = "Infertility (1 = Yes, 0 = No)") +  
theme_minimal()`



## Histogram: Distribution of Age:

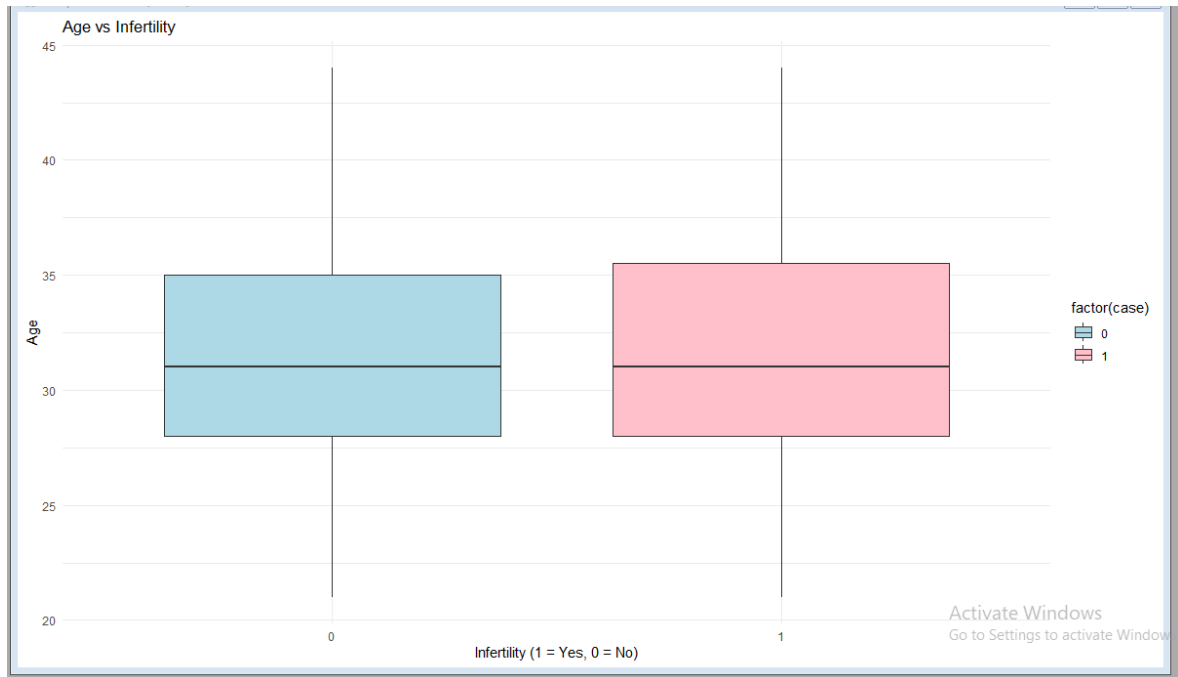
To understand the age distribution of women in the dataset

```
ggplot(I, aes(x = age)) +  
  geom_histogram(bins = 20, fill = "blue", color = "white", alpha = 0.7) +  
  labs(title = "Age Distribution of Women",  
        x = "Age",  
        y = "Frequency") +  
  theme_minimal()
```



## Boxplot: Age vs. Infertility (case):

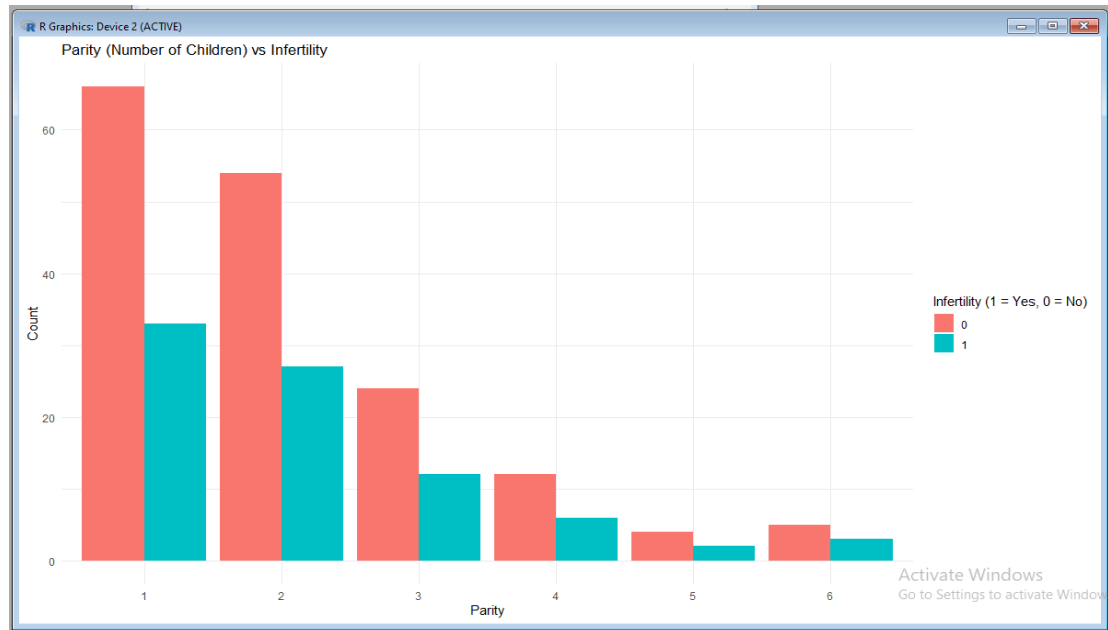
This shows how the age of women varies based on whether they experienced infertility (case)



## Bar Plot: Parity (Number of Children) by Infertility Case:

This plot shows how the number of children (parity) relates to infertility (case).

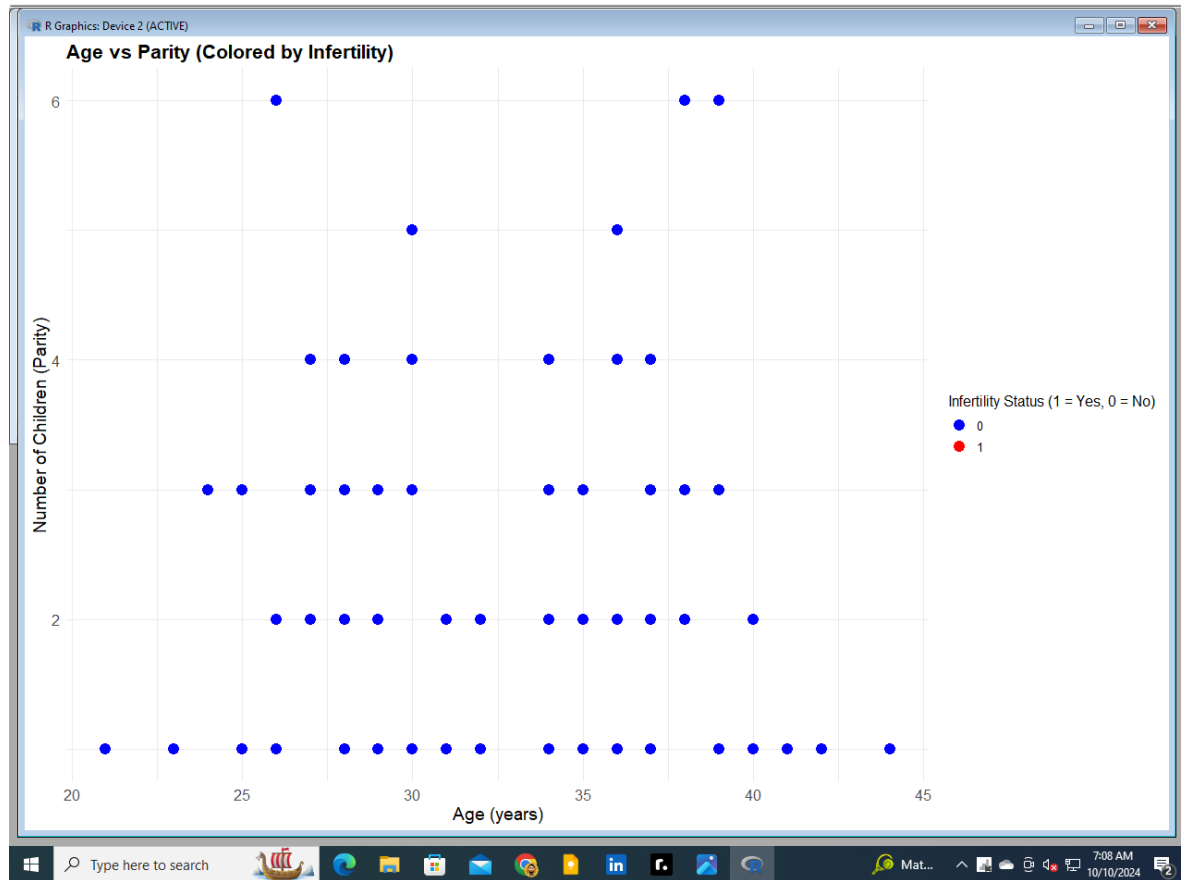
```
ggplot(I, aes(x = factor(parity), fill = factor(case))) +  
geom_bar(position = "dodge") +  
labs(title = "Parity (Number of Children) vs Infertility",  
x = "Parity",  
y = "Count",  
fill = "Infertility (1 = Yes, 0 = No)") +  
theme_minimal()
```



## Scatter Plot: Age vs Parity, Colored by Infertility (case)

This scatter plot helps visualize the relationship between age and parity, with points colored based on the infertility status.

```
ggplot(I, aes(x = age, y = parity, color = factor(case))) +
+   geom_point(size = 4) + # Adjust the size of points
+   labs(
+     title = "Age vs Parity (Colored by Infertility)", # Main title
+     x = "Age (years)", # X-axis label
+     y = "Number of Children (Parity)", # Y-axis label
+     color = "Infertility Status (1 = Yes, 0 = No)" # Legend title
+   ) +
+   theme_minimal() + # I want to customize this theme to test my ability of
+   # customization theme
+   theme(
+     plot.title = element_text(size = 16, face = "bold"), # Title size and bold text
+     axis.title.x = element_text(size = 14), # X-axis title size
+     axis.title.y = element_text(size = 14), # Y-axis title size
+     axis.text = element_text(size = 12), # Axis text size
+     legend.title = element_text(size = 12), # Legend title text size
+     legend.text = element_text(size = 10), # Legend text size
+   ) +
+   scale_color_manual(values = c("blue", "red"))
```

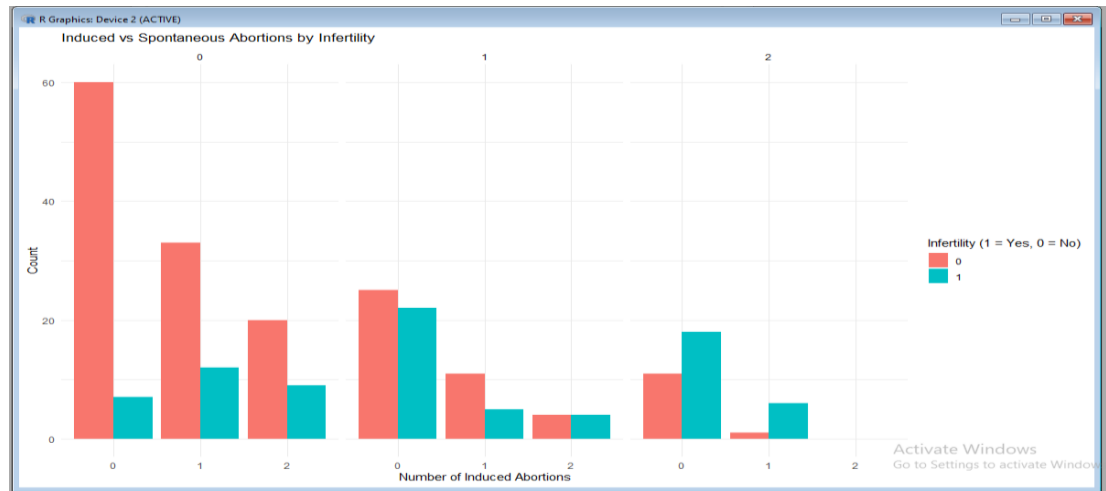


## Faceted Bar Plot: Induced and Spontaneous Abortions by Infertility

This plot shows the relationship between induced and spontaneous abortions with infertility, faceted by the number of spontaneous abortions.

```
ggplot(I, aes(x = factor(induced), fill = factor(case))) +
  geom_bar(position = "dodge") +
  facet_wrap(~spontaneous) +
  labs(title = "Induced vs Spontaneous Abortions by Infertility",
        x = "Number of Induced Abortions",
        y = "Count",
        fill = "Infertility (1 = Yes, 0 = No)") +
  theme_minimal()
```

**Note:** Except for a few I am not customizing themes for all visuals intentionally because it is not an assignment or project task in which mention a special theme and label font size etc. compulsory. The aim of this work is to show and examine my ability in R programing by doing different analysis.



## Pair Plot (Relationships Between Multiple Variables)

To visualize pairwise relationships between multiple variables like age, parity, and education, we can use the GGally package.

```
install.packages("GGally")
```

```
library(GGally)
```

```
ggpairs(I, columns = c("age", "parity", "education", "induced"), aes(color = factor(case)))
```





*Calculate the Percentage Increase in Infertility Risk:*

```
library(dplyr)
I <- I %>%
  mutate(age_group = case_when(
    age < 30 ~ "Under 30",
    age >= 30 & age < 40 ~ "30-39",
    age >= 40 ~ "40 and above"
  ))
baseline_risk <- I %>%
  group_by(age_group) %>%
  summarise(baseline_inf_rate = mean(case == 1) * 100)
post_abortion_risk <- I %>%
  filter(induced == 1) %>%
  group_by(age_group) %>%
  summarise(post_abortion_inf_rate = mean(case == 1) * 100)
risk_comparison <- baseline_risk %>%
  left_join(post_abortion_risk, by = "age_group") %>%
  mutate(percentage_increase = ((post_abortion_inf_rate - baseline_inf_rate) /
    baseline_inf_rate) * 100)
```

```
# Display the results  
print(risk_comparison)
```

Age group	Base line inf. rate	Post abortion inf. rate	%_increase
30-39	33.6	33.3	-0.755
40 and above	33.3	50	50
Under 30	33.3	32.4	-2.94

## Advance Analysis:

### Build a Decision Tree:

We can use a decision tree to model the relationship between the variables in our dataset, including predicting the likelihood of infertility (case) based on factors like spontaneous, induced, age, and others. As we know that Decision Trees are useful for classification tasks like this, where we want to predict a binary outcome (infertility: 1 or 0).

```
# Load necessary libraries
```

```
library(rpart) # Necessary for building decision tree
```

```
library(rpart.plot) # And this one is for visualizing the decision tree
```

```
# Ensure the 'case' column is a factor (binary outcome variable)
```

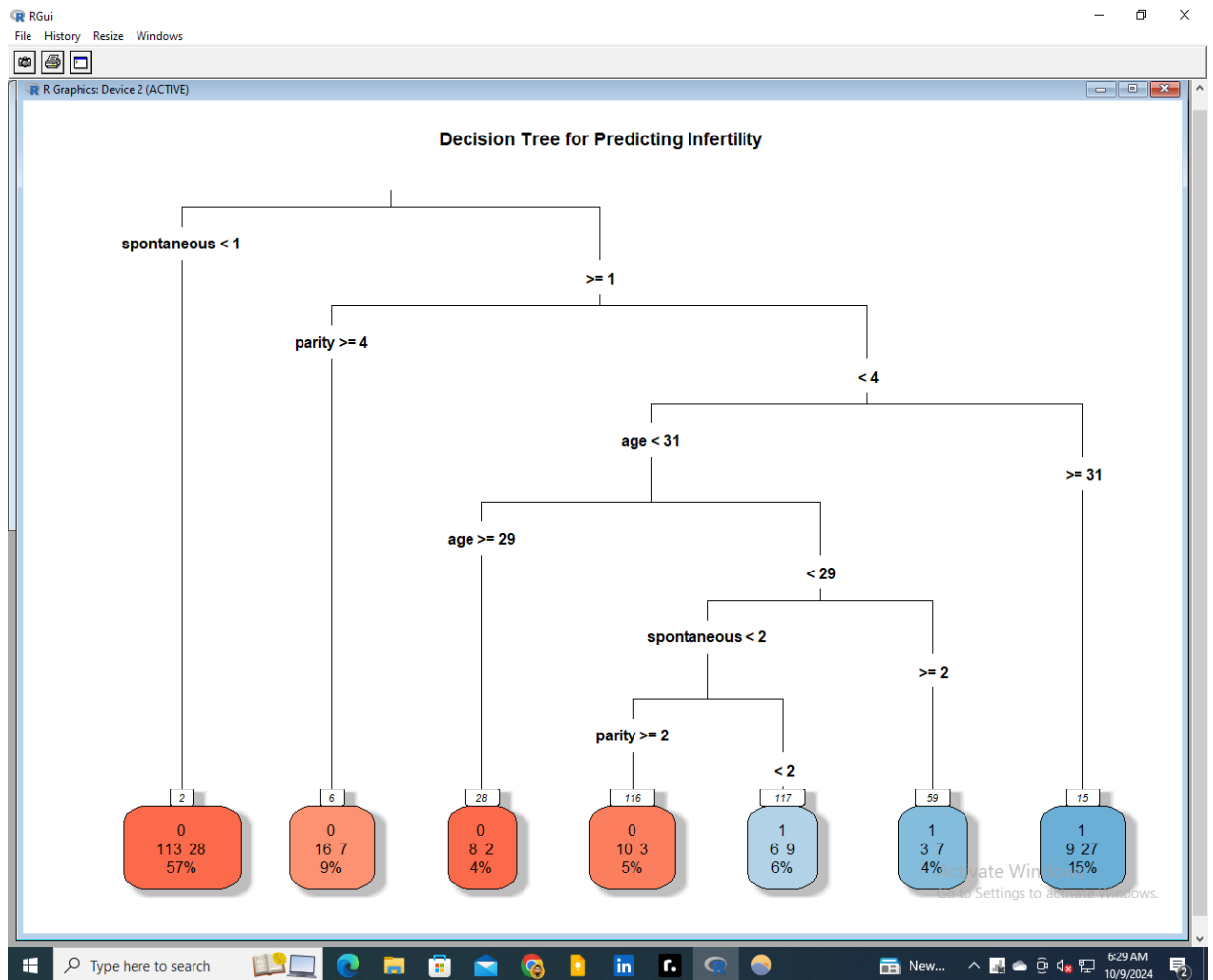
```
I$case <- as.factor(I$case)
```

```
# we use variables for Predict 'case' (infertility) using 'age', 'parity', 'spontaneous',  
'induced', and other relevant variables so,
```

```
tree_model <- rpart(case ~ age + parity + spontaneous + induced + education, data = I,  
method = "class")
```

```
# Visualize the decision tree
```

```
rpart.plot(tree_model, type = 3, extra = 101, fallen.leaves = TRUE, main = "Decision  
Tree for Predicting Infertility", box.palette = "RdBu", shadow.col = "gray", nn = TRUE)
```



### Evaluate the Model:

we can evaluate the performance of our decision tree using the following code

```

predictions <- predict(tree_model, I, type = "class")
# Confusion matrix to evaluate the model
table(Predicted = predictions, Actual = I$case)
# Calculate accuracy
accuracy <- sum(predictions == I$case) / nrow(I)
cat("Accuracy of the decision tree:", accuracy * 100, "%\n")

```

This will give us an idea of how well the decision tree is performing in terms of predicting infertility based on the given factors.

We can adjust the variables in the model depending on which ones we think are most relevant.

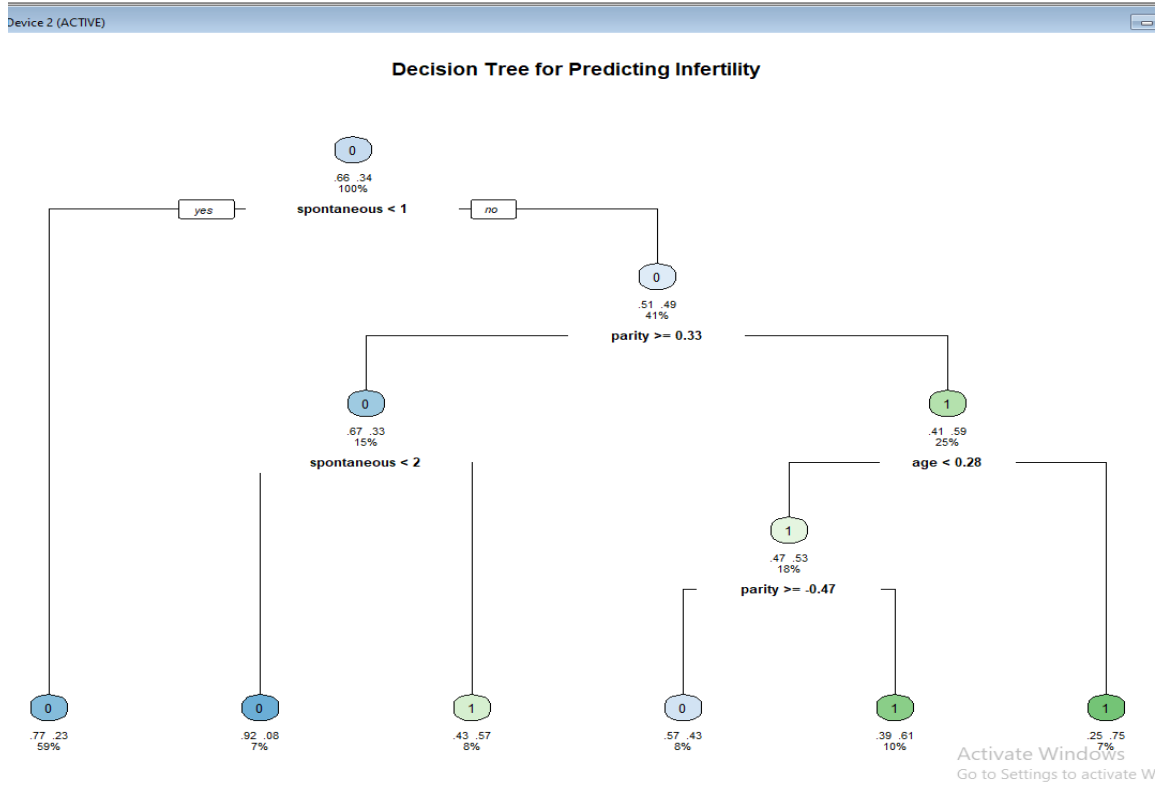
If we have more data, it's a good idea to split the dataset into training and testing sets to evaluate the performance of the decision tree on unseen data.

```
> # Confusion matrix to evaluate the model
> table(Predicted = predictions, Actual = I$case)
      Actual
Predicted 0    1
0      147   40
1       18   43
>
> # Calculate accuracy
> accuracy <- sum(predictions == I$case) / nrow(I)
> cat("Accuracy of the decision tree:", accuracy * 100, "%\n")
Accuracy of the decision tree: 76.6129 %
1
```

Hmm! I think the screenshot looks clear; can you see?

*We also try 2nd method to make Decision Tree model:*

```
# Train the Decision Tree model
tree_model <- rpart(case ~ age + parity + spontaneous + induced + education,
data = train_data, method = "class")
# Predict on the test data
tree_preds <- predict(tree_model, test_data, type = "class")
# Confusion matrix and accuracy
confusionMatrix(tree_preds, test_data$case)
Visualize Decision Tree
library(rpart.plot)
# Plot the decision tree
rpart.plot(tree_model, type = 2, extra = 104, under = TRUE,
main = "Decision Tree for Predicting Infertility",
fallen.leaves = TRUE, cex = 0.8)
```



## Build a Neural Network:

We definitely build Neural Network (NN) models in R using this dataset. Neural networks can be used for classification tasks such as predicting infertility (binary outcome) based on variables like age, parity, spontaneous, induced, and others.

R is not stereotypical language it is advance one it provides us various libraries to build neural networks. First of all we install them.

```
install.packages("nnet")
install.packages("caret")
install.packages("NeuralNetTools")
```

After loading libraries

Neural networks work better when the data is normalized, so we will scale continuous variables (age, parity) and convert 'case' to factor

```
I$age <- scale(I$age)
I$parity <- scale(I$parity)
I$case <- as.factor(I$case)
```

Split the Data into Training and Testing Sets

```
set.seed(123) # Set seed for reproducibility
train_index <- createDataPartition(I$case, p = 0.7, list = FALSE) # 70% training
train_data <- I[train_index, ]
test_data <- I[-train_index, ]
```

- Now we Build and Train the Neural Network:

```
# Predict 'case' using 'age', 'parity', 'spontaneous', and 'induced'

nn_model <- nnet(case ~ age + parity + spontaneous + induced,
data = train_data,

size = 5,    # Number of neurons in the hidden layer

decay = 0.01, # Regularization parameter to prevent overfitting

maxit = 200) # Maximum number of iterations

# Print model summary

summary(nn_model)
```

- Then Evaluate the Neural Network on Test Data

```
# Make predictions on the test data

predictions <- predict(nn_model, test_data, type = "class")

# Confusion matrix to evaluate performance

confusion_matrix <- table(Predicted = predictions, Actual = test_data$case)

# Calculate accuracy

accuracy <- sum(predictions == test_data$case) / nrow(test_data)

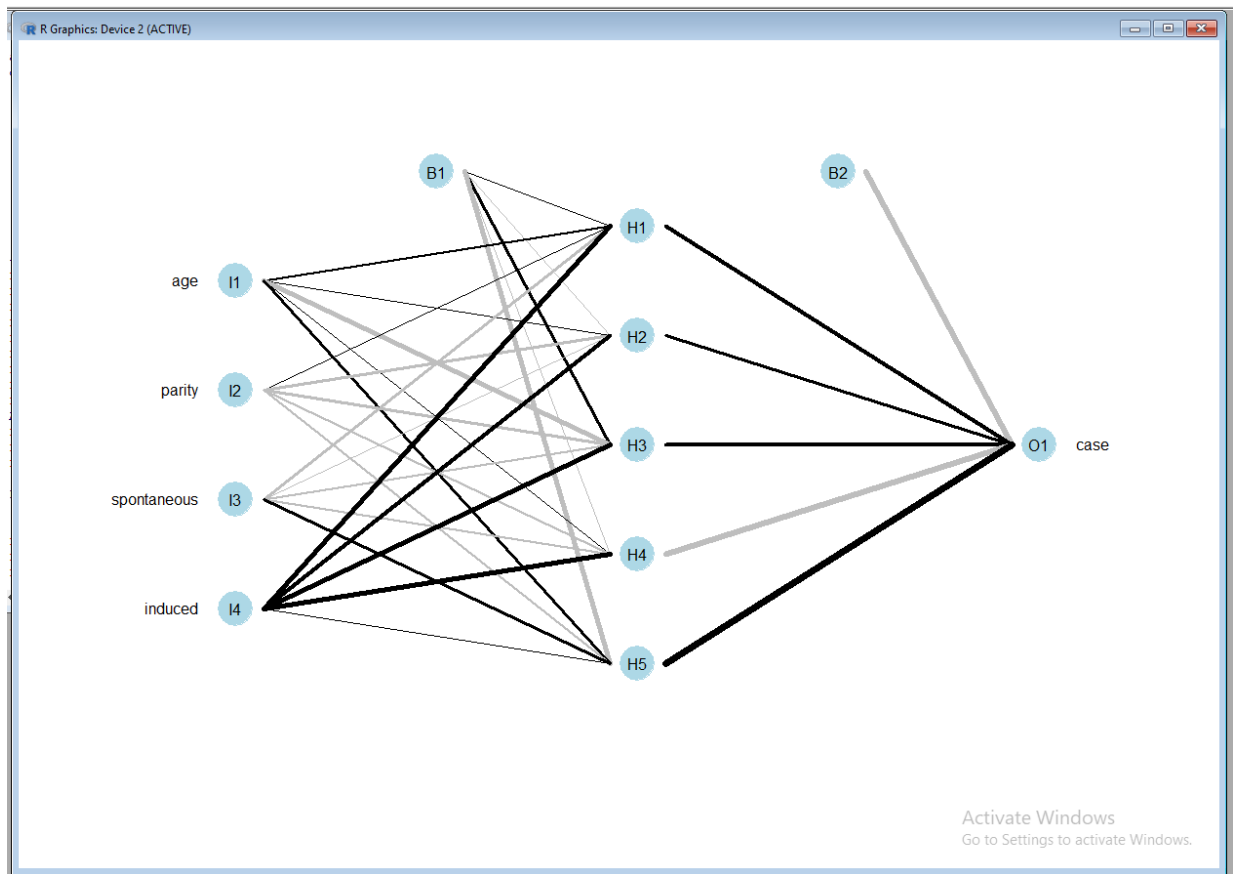
cat("Accuracy of the neural network:", accuracy * 100, "%\n")

# Display the confusion matrix

print(confusion_matrix)
```

- Visualize the Neural Network

```
plotnet(nn_model)
```



## Logistic Regression

Now we fit the logistic regression model to predict case (infertility) based on predictors like age and parity.

### *Visualization of Predicted Probability by Age:*

First, we will ensure the data is ready for logistic regression by converting categorical variables into numeric form.

**# Load necessary libraries**

```
library(ggplot2)
```

```
library(dplyr)
```

**# Ensure categorical variables are converted to numeric**

```
I_numeric <- I
```

**# Convert 'education' to numeric**

```

I_numeric$education <- as.numeric(as.factor(I_numeric$education))

# Convert 'induced', 'spontaneous', and 'case' to numeric

I_numeric$induced <- as.numeric(I_numeric$induced)

I_numeric$spontaneous <- as.numeric(I_numeric$spontaneous)

I_numeric$case <- as.numeric(I_numeric$case) - 1 # Make case binary: 0 and 1

# Logistic regression model: predicting infertility (case) based on age and parity

logistic_model <- glm(case ~ age + parity + induced + spontaneous,
data = I_numeric,

family = binomial)

summary(logistic_model)

```

```

Call:
glm(formula = case ~ age + parity + induced + spontaneous, family = binomial,
    data = I_numeric)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.657e+01  1.471e+05      0      1
age          2.519e-15  4.492e+03      0      1
parity      -2.324e-14  2.405e+04      0      1
induced       1.057e-14  4.027e+04      0      1
spontaneous  -6.949e-15  3.804e+04      0      1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 0.0000e+00  on 247  degrees of freedom
Residual deviance: 1.4388e-09  on 243  degrees of freedom
AIC: 10

Number of Fisher Scoring iterations: 25

> |

```

We will now generate the predicted probabilities for each observation based on the logistic model.

```

I_numeric$predicted_probs <- predict(logistic_model, type = "response")

# Load the ggplot2 library

library(ggplot2)

# Plot Age vs Predicted Probability of Infertility

ggplot(I, aes(x = age, y = predicted_probs, color = as.factor(case))) +

```



```
geom_point(size = 3) +

  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = FALSE, color =
"blue") +

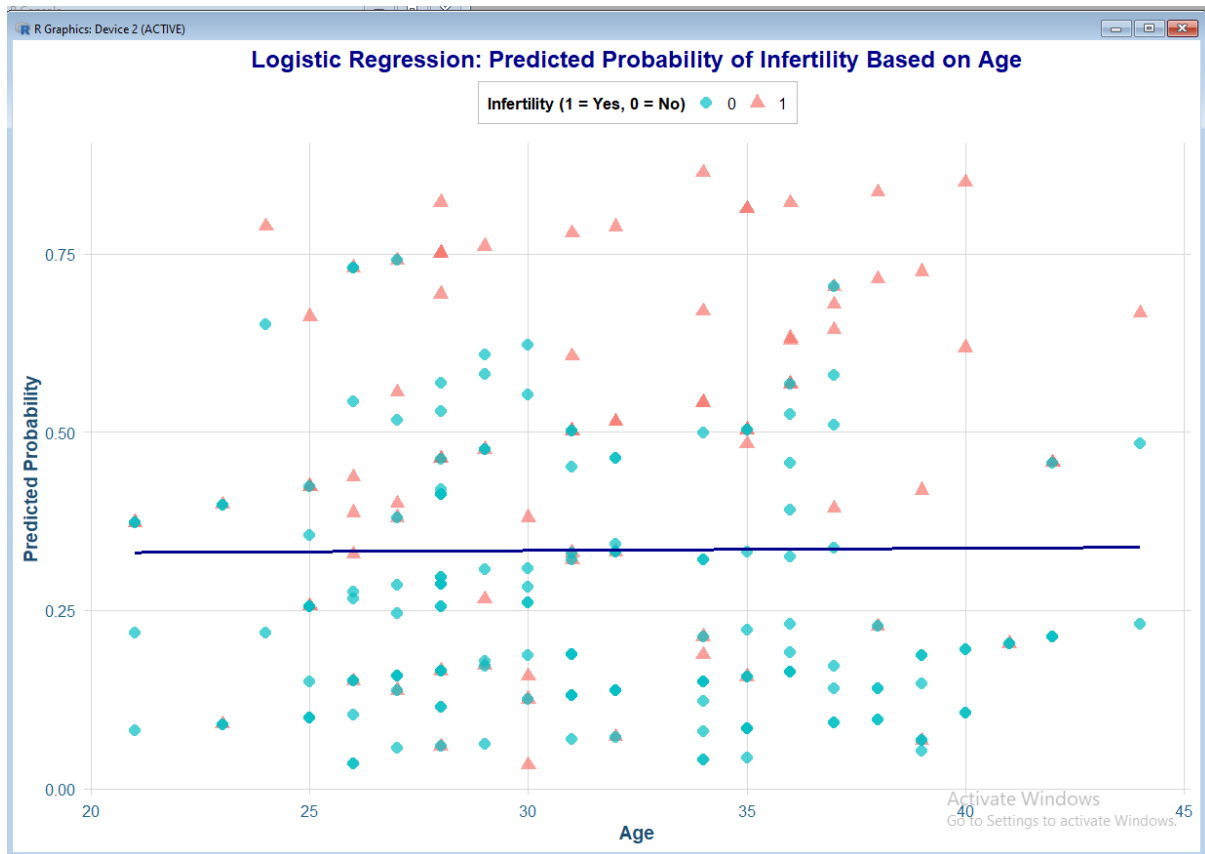
  labs(title = "Predicted Probability of Infertility by Age",

x = "Age",

y = "Predicted Probability of Infertility",

color = "Infertility (1 = Yes, 0 = No)") +

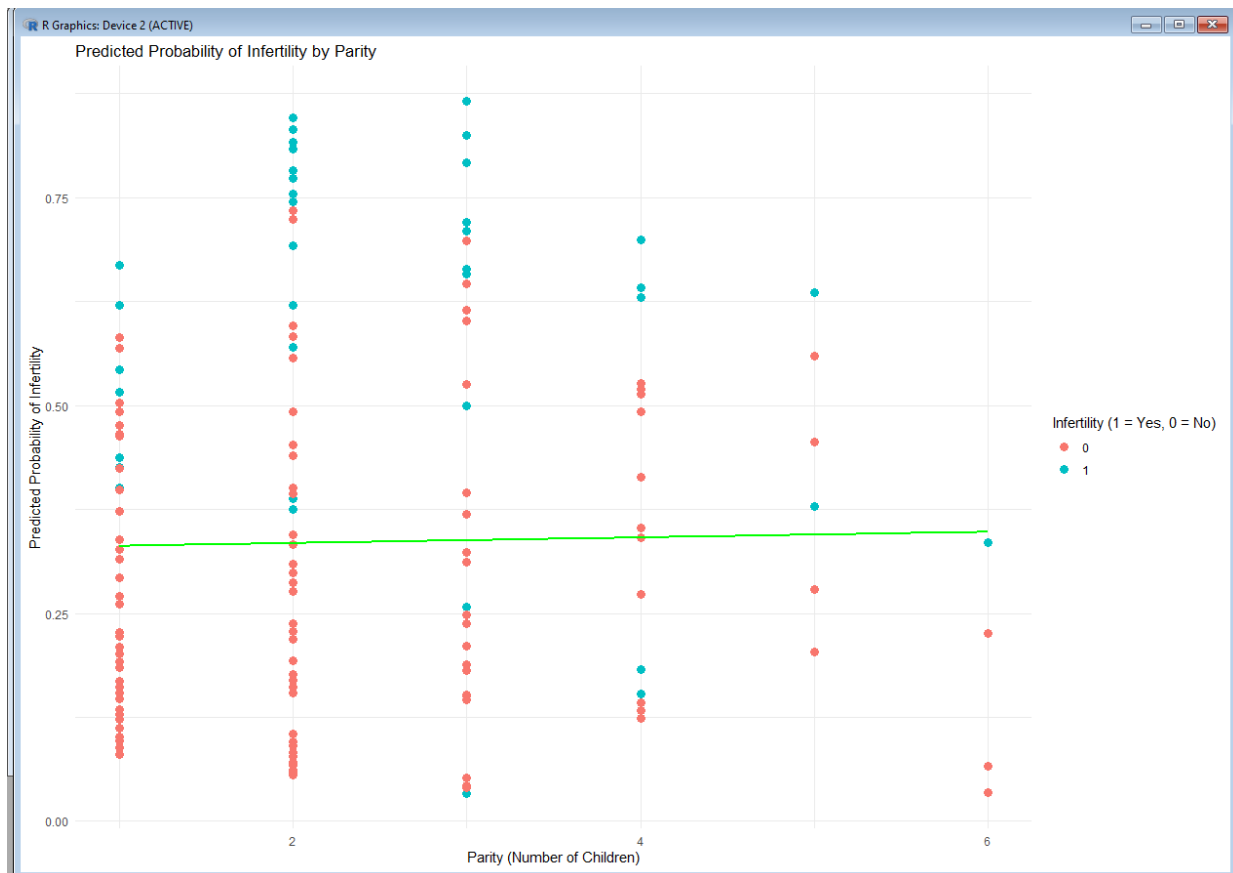
theme_minimal()
```



Visualization of Predicted Probability by Parity:

```
ggplot(I, aes(x = parity, y = predicted_probs, color = as.factor(case))) +
geom_point(size = 3) +
geom_smooth(method = "glm", method.args = list(family = "binomial"), se = FALSE, color =
"green") +
  labs(title = "Predicted Probability of Infertility by Parity",
x = "Parity (Number of Children)",
```

```
y = "Predicted Probability of Infertility",
color = "Infertility (1 = Yes, 0 = No)" +
theme_minimal()
```



To visualize the logistic regression results, we can plot the predicted probabilities against one of the key variables, such as age or parity. Below is the code to visualize the logistic regression model.

```
# Generate predicted probabilities based on the model
```

```
I_numeric$predicted_probs <- predict(logistic_model, type = "response")
```

```
library(ggplot2)
```

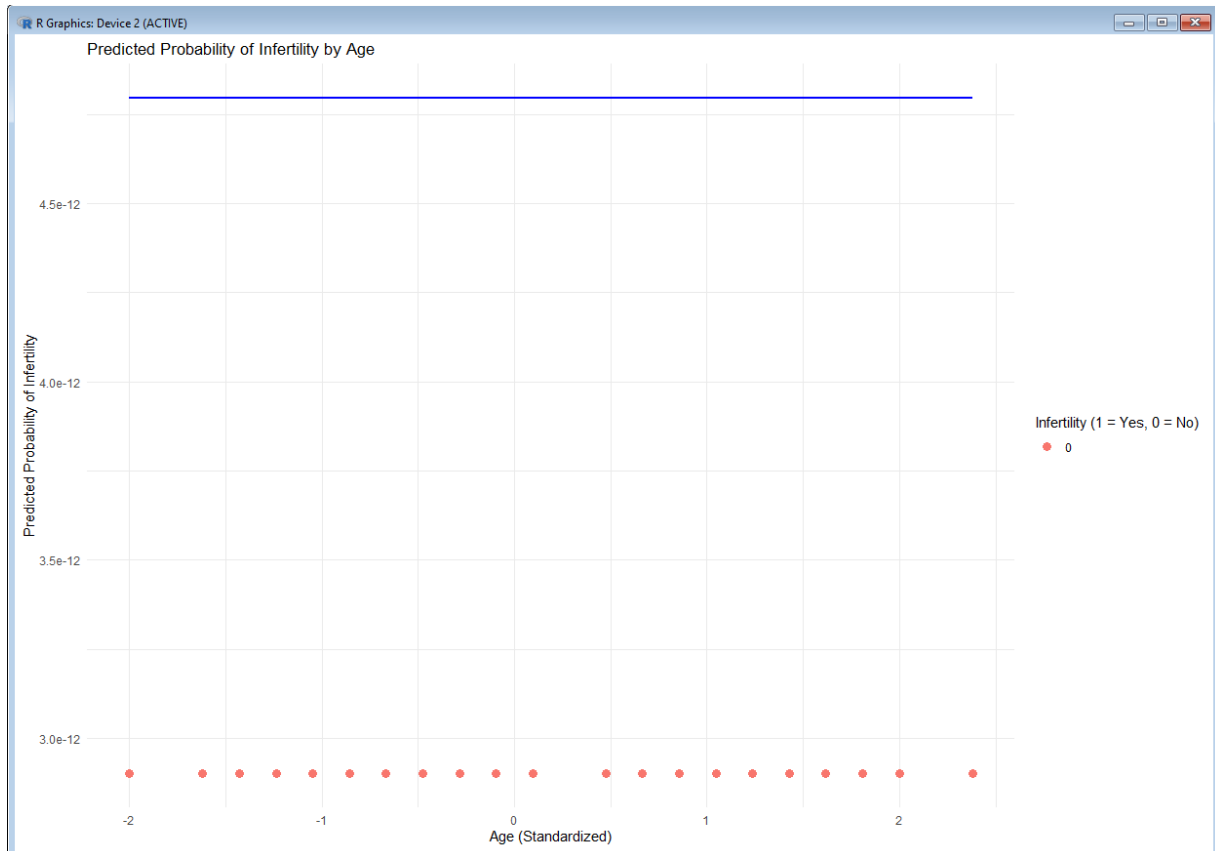
```
# Plot Age vs Predicted Probability of Infertility
```

```
ggplot(I_numeric, aes(x = age, y = predicted_probs, color = as.factor(case))) +
```

```
geom_point(size = 3) + # Scatter plot of age and predicted probabilities
```

```
geom_smooth(method = "glm", method.args = list(family = "binomial"), se = FALSE, color =
"blue") + # Logistic regression line
```

```
labs(title = "Predicted Probability of Infertility by Age",  
x = "Age (Standardized)",  
y = "Predicted Probability of Infertility",  
color = "Infertility (1 = Yes, 0 = No)") +  
theme_minimal()
```



### *Instability Issue*

R console show this image with the warning message **"Warning message: glm.fit: algorithm did not converge"**

It means that the logistic regression model couldn't find a stable solution. In logistic regression, the algorithm tries to estimate the relationship between the predictor variables and the outcome variable (in our case, infertility).

### **Common Reasons of this warning:**

**Multicollinearity:** High correlation between predictor variables makes it difficult to separate their effects.

**Outliers or Extreme Values:** Unusually large or small values in the data can cause instability.

**Complete Separation:** If the predictor variables perfectly predict the outcome (e.g., all cases of infertility happen for specific ages), the algorithm may struggle.

**Insufficient Variation:** Not enough variability in the outcome variable (e.g., mostly 0s or mostly 1s).

It's important to remember that **data analysis is not always about finding perfect or stable models**. Often, **real-world data is messy and complex**, especially in fields like health and biology where variability is natural. So worry about instability in pattern.

**The question is “How to Fix It”:**

**There are three solutions**

- Check for multicollinearity.
- Remove or adjust extreme outliers.
- Try simplifying the model by removing less important predictors.

I am just trying to find the solution

**Check for Multicollinearity**

Multicollinearity can cause instability in logistic regression. We'll use Variance Inflation Factor (VIF) to detect multicollinearity and remove highly correlated predictors.

*# Install and load necessary packages*

```
install.packages("car")
```

```
library(car)
```

*# Check for multicollinearity using VIF*

```
vif_values <- vif(glm(case ~ age + parity + induced + spontaneous, data = I, family = binomial))
```

*# Print VIF values*

```
print(vif_values)
```

*# Typically, VIF > 5 or 10 indicates multicollinearity*

*# Remove the predictor with the highest VIF if necessary*

*Remove or Adjust Extreme Outliers*

We'll visualize outliers and either remove or adjust them.

```
# Visualize outliers in the data using boxplot for each numeric variable
```

```
boxplot(l$age, main = "Boxplot of Age", col = "lightblue")
```

```
boxplot(l$parity, main = "Boxplot of Parity", col = "lightblue")
```

```
# Optionally, we can remove outliers if they are extreme
```

```
# Define a function to remove outliers
```

```
remove_outliers <- function(x, na.rm = TRUE, ...) {
```

```
  qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm, ...)
```

```
  H <- 1.5 * IQR(x, na.rm = na.rm)
```

```
  y <- x
```

```
  y[x < (qnt[1] - H)] <- NA
```

```
  y[x > (qnt[2] + H)] <- NA
```

```
  return(y)
```

```
}
```

```
# Apply the function to the numeric columns
```

```
l$age <- remove_outliers(l$age)
```

```
l$parity <- remove_outliers(l$parity)
```

```
# Remove rows with NA values (outliers)
```

```
l_clean <- na.omit(l)
```

### **Simplify the Model by Removing Less Important Predictors**

If the model is too complex, we try to removing some of the predictors.

```
# Fit a simpler logistic regression model with fewer predictors
```

```
logistic_model_simple <- glm(case ~ age + parity, data = l_clean, family = binomial)
```

```
# Check summary to see if the algorithm converged
```

```
summary(logistic_model_simple)
```

### *Visualizing the Logistic Regression*

Once the model has been simplified, cleaned of multicollinearity, and outliers, So I can visualize the logistic regression results. Here's how to do it:

#### Plotting Predicted Probabilities from Logistic Regression

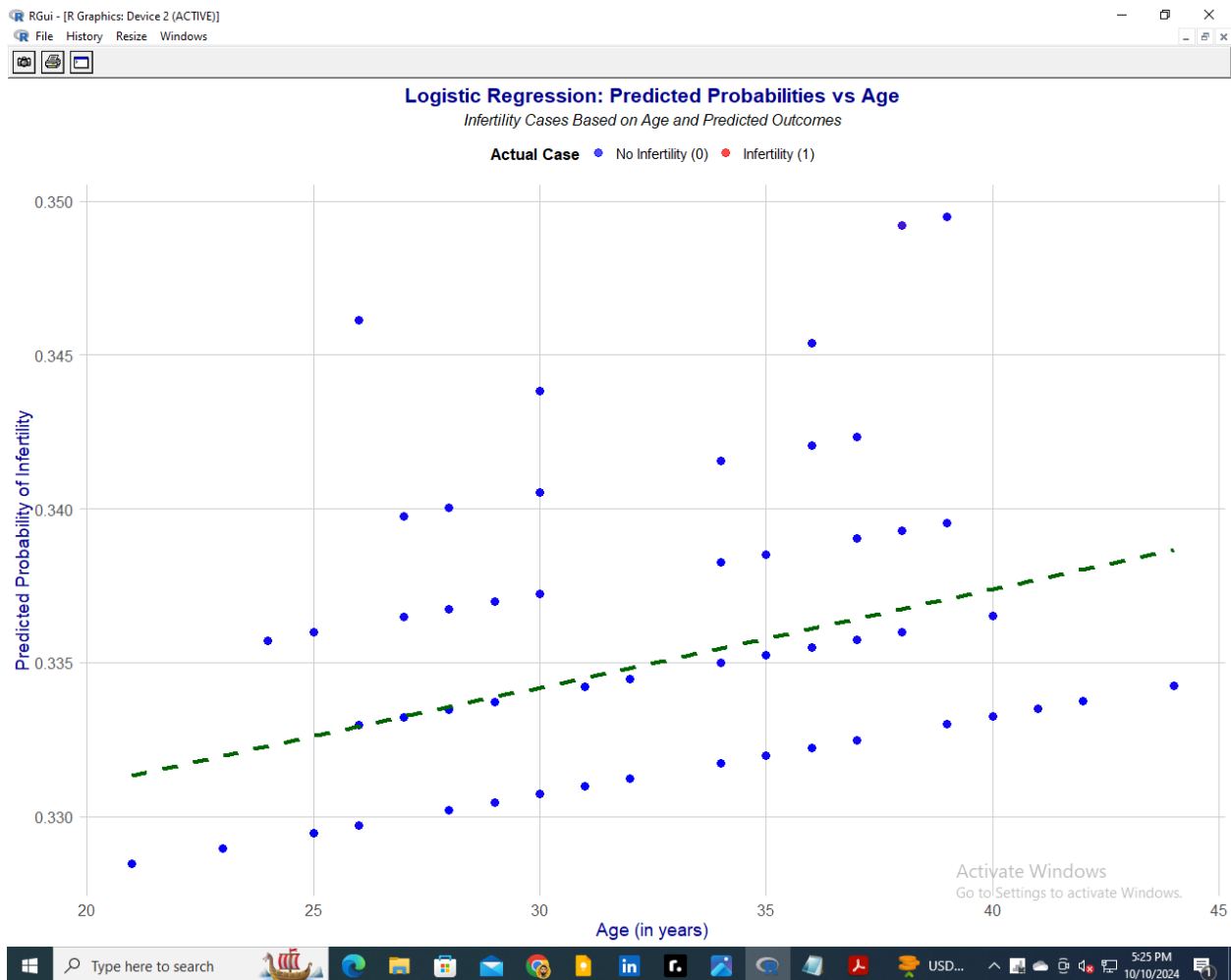
```
# Predict probabilities from the logistic model

l_clean$predicted_probs <- predict(logistic_model_simple, type = "response")

# Plot actual cases vs predicted probabilities

library(ggplot2)

ggplot(l_clean, aes(x = age, y = predicted_probs, color = as.factor(case))) +
  geom_point(size = 3) +
  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = FALSE) +
  labs(title = "Logistic Regression: Predicted Probabilities vs Age",
       x = "Age",
       y = "Predicted Probability of Infertility (Case)",
       color = "Actual Case (0 = No, 1 = Yes)") +
  theme_minimal() +
  scale_color_manual(values = c("blue", "red")) # Customize colors
```



I received result with this message:

Warning message:

In eval(family\$initialize) : non-integer #successes in a binomial glm!

## The k-nearest neighbors (KNN) algorithm:

The k-Nearest Neighbors (KNN) algorithm is a simple and powerful classification algorithm. Since we are working with the I dataset (Infertility data), I'll show you how to apply KNN for predicting case (infertility) and visualize it with a customized theme. First of all

# Install necessary libraries

```
install.packages(c("class", "caret", "ggplot2", "scales"))
```

```

library(class) # For KNN

library(caret) # For data splitting and confusion matrix

library(ggplot2) # For visualization

library(scales) # For customizing color scales

# Scaling the numeric variables

scaled_data <- I # Copy the data

scaled_data$age <- scale(I$age) # Scale age

scaled_data$parity <- scale(I$parity) # Scale parity

scaled_data$spontaneous <- scale(I$spontaneous) # Scale spontaneous abortions

scaled_data$induced <- scale(I$induced) # Scale induced abortions


# Split the data into training and testing sets

set.seed(123) # For reproducibility

trainIndex <- createDataPartition(scaled_data$case, p = 0.7, list = FALSE)

train_data <- scaled_data[trainIndex, ]

test_data <- scaled_data[-trainIndex, ]


# Define predictor variables (independent) and target variable (dependent)

train_x <- train_data[, c("age", "parity", "spontaneous", "induced")]

train_y <- train_data$case


test_x <- test_data[, c("age", "parity", "spontaneous", "induced")]

test_y <- test_data$case

# Apply the KNN algorithm (k = 5)

knn_predictions <- knn(train = train_x, test = test_x, cl = train_y, k = 5)

knn_predictions <- as.factor(knn_predictions)

test_y <- as.factor(test_y)

confusionMatrix(knn_predictions, test_y)

```



```
test_data$predicted_class <- as.numeric(knn_predictions) - 1 # Convert predictions to 0/1 for
visualization
```

```
test_data$actual_class <- test_y
```

```
# Visualization: Age vs Parity colored by predicted infertility class
```

```
ggplot(test_data, aes(x = age, y = parity, color = as.factor(predicted_class), shape =
as.factor(actual_class))) +
```

```
  geom_point(size = 3, alpha = 0.7) +
```

```
  labs(title = "KNN: Predicted Infertility (Case) based on Age and Parity",
  subtitle = "Comparing Actual vs Predicted Cases",
```

```
  x = "Age (Scaled)",
```

```
  y = "Parity (Scaled)",
```

```
  color = "Predicted Class (0 = No, 1 = Yes)",
```

```
  shape = "Actual Class (0 = No, 1 = Yes)") +
```

```
# Custom color scale and point shapes
```

```
  scale_color_manual(values = c("blue", "red")) +
```

```
  scale_shape_manual(values = c(16, 17)) + # Different shapes for actual class
```

```
  theme_minimal() +
```

```
  theme(
```

```
    plot.title = element_text(size = 16, face = "bold", color = "darkblue", hjust = 0.5),
```

```
    plot.subtitle = element_text(size = 12, face = "italic", hjust = 0.5),
```

```
    axis.title.x = element_text(size = 14, color = "darkblue"),
```

```
    axis.title.y = element_text(size = 14, color = "darkblue"),
```

```
    axis.text = element_text(size = 12),
```

```
    legend.title = element_text(size = 13, face = "bold"),
```

```
    legend.text = element_text(size = 11),
```

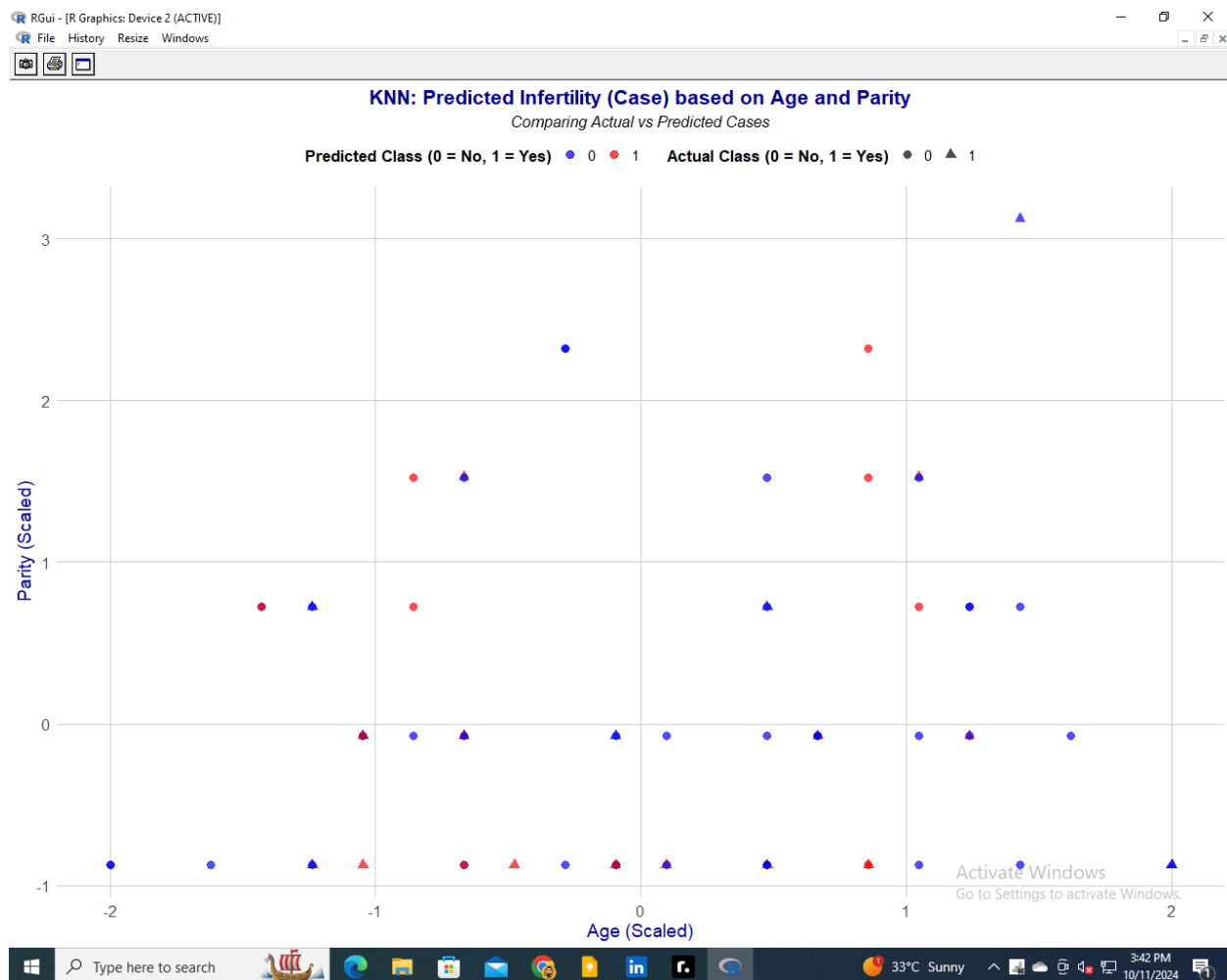
```
panel.grid.major = element_line(color = "gray80"), # Light grid lines
```

```
panel.grid.minor = element_blank(), # Remove minor grid lines
```

```
legend.position = "top", # Move legend to the top
```

```
legend.direction = "horizontal"
```

```
)
```



```
# Predict on the test data
```

```
rf_preds <- predict(rf_model, test_data)
```

```
# Confusion matrix and accuracy
```

```
confusionMatrix(rf_preds, test_data$case)
```

As we see that this project objective is to examine and show programming skill by doing different tasks so I want to perform an algorithm SVM is another powerful algorithm for binary classification, which works well when the data is separable.

*Support Vector Machine (SVM)*

```
# Train the SVM model
```

```
svm_model <- svm(case ~ age + parity + spontaneous + induced + education, \
```

```
data = train_data, kernel = "linear", cost = 1)
```

```
# Predict on the test data
```

```
svm_preds <- predict(svm_model, test_data)
```

```
confusionMatrix(svm_preds, test_data$case)
```

```
Confusion Matrix and Statistics

      Reference
Prediction 0  1
0  39 10
1  10 14

      Accuracy : 0.726
      95% CI : (0.6091, 0.8239)
      No Information Rate : 0.6712
      P-Value [Acc > NIR] : 0.1926

      Kappa : 0.3793

      Mcnemar's Test P-Value : 1.0000

      Sensitivity : 0.7959
      Specificity : 0.5833
      Pos Pred Value : 0.7959
      Neg Pred Value : 0.5833
      Prevalence : 0.6712
      Detection Rate : 0.5342
      Detection Prevalence : 0.6712
      Balanced Accuracy : 0.6896

      'Positive' Class : 0

> |
```

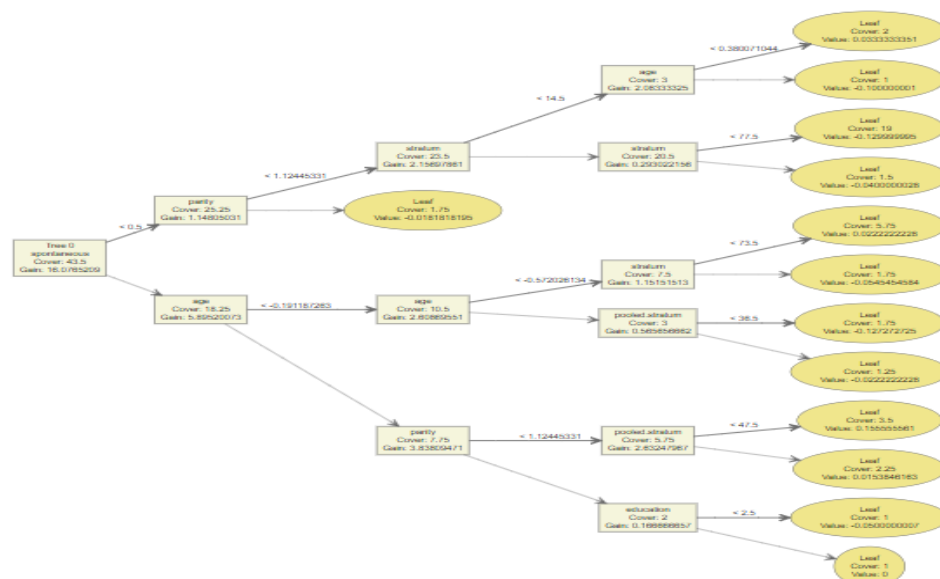
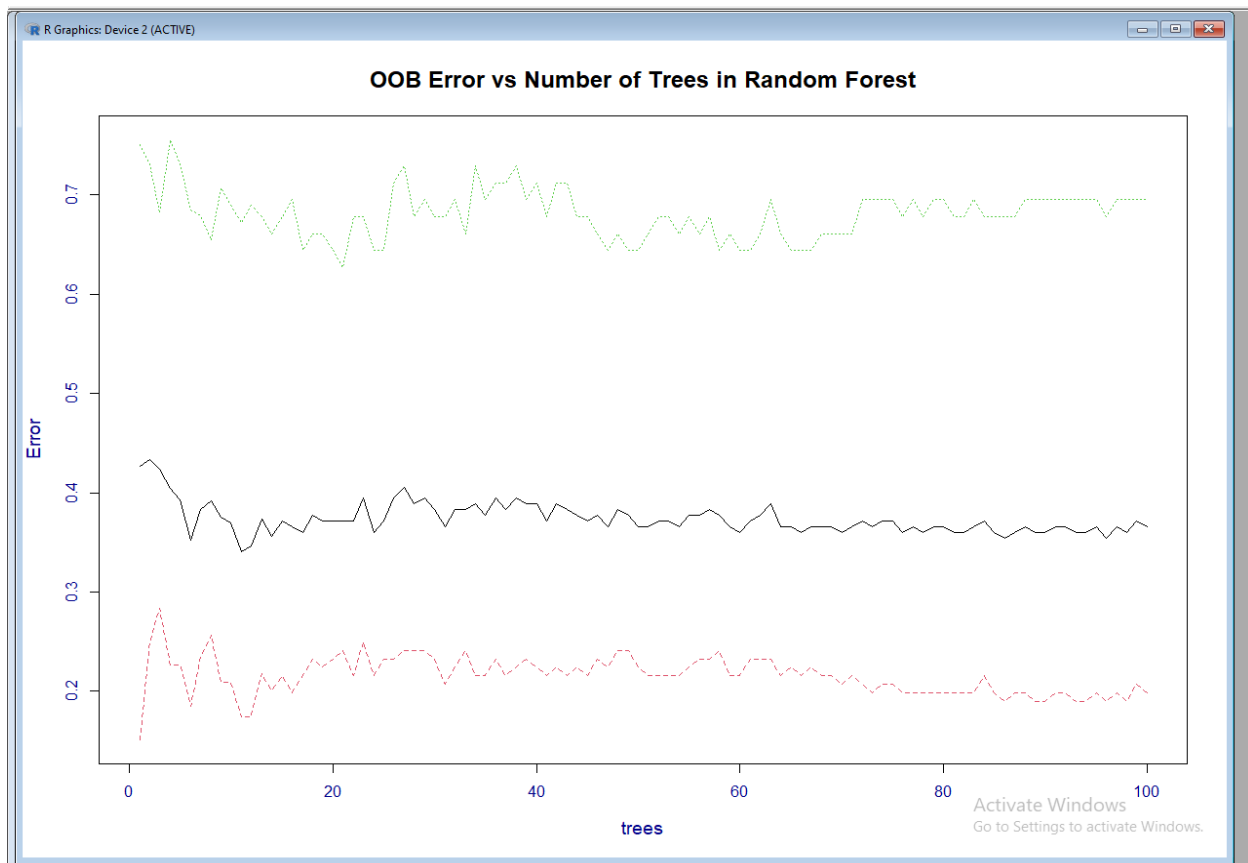
Visualize Random forest

```
# Plot variable importance
```

```
varImpPlot(rf_model, main = "Variable Importance in Random Forest")
```

```
# Plot OOB error progression (as the number of trees increases)
```

```
plot(rf_model, main = "OOB Error vs Number of Trees in Random Forest")
```



## Visualization of Random Forest Variable Importance:

# Load required library

```
library(randomForest)
```

# Set plot parameters for improved visualization (for base R plots)

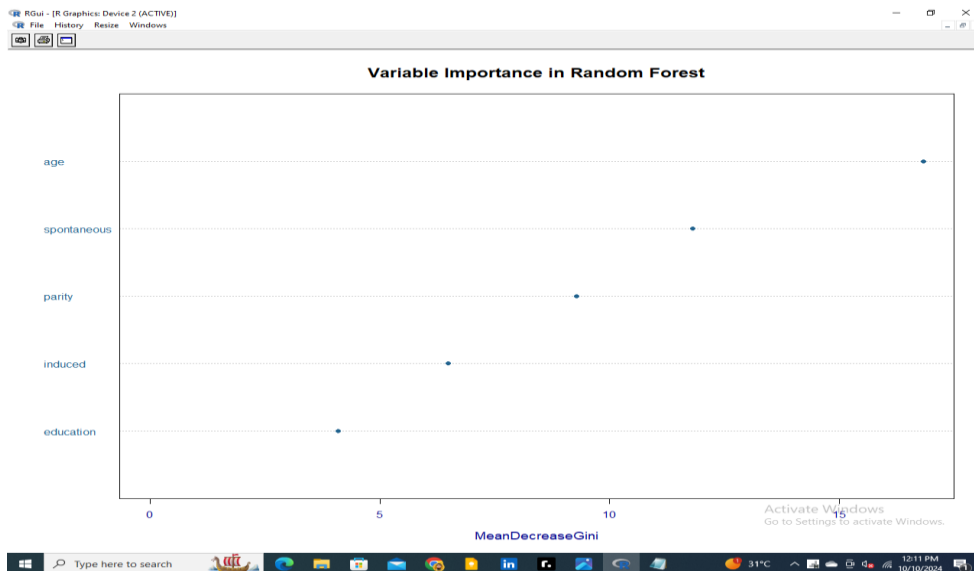
```
par(mfrow = c(1, 1), # Set to a single plot per figure
```

```
    mar = c(5, 4, 4, 2) + 0.1, # Set margins + cex.main = 1.5, # Title font size + cex.lab = 1.2,
```

```
    cex.axis = 1, # Axis tick font size + col.lab = "darkblue", + col.axis = "darkblue", + font.main =  
    2) varImpPlot(rf_model,
```

```
    main = "Variable Importance in Random Forest", # Title
```

```
    pch = 16, # Use filled circle points + col = "#1F618D")
```



## XGBoost (Extreme Gradient Boosting)

XGBoost is one of the most powerful algorithms for classification and can handle complex datasets very well.

# Prepare data for XGBoost (matrix form and numeric labels)

```
train_matrix <- as.matrix(train_data[, -which(names(train_data) == "case")])
```

```
test_matrix <- as.matrix(test_data[, -which(names(test_data) == "case")])
```

```
train_labels <- as.numeric(train_data$case) - 1 # Convert factor levels to 0 and 1
```

```
test_labels <- as.numeric(test_data$case) - 1
```

```
xgb_model <- xgboost(data = train_matrix, label = train_labels,
```

```
max_depth = 4, eta = 0.1, nrounds = 100,
```

```
objective = "binary:logistic")
```

```
xgb_preds <- predict(xgb_model, test_matrix)
```

```
xgb_class <- ifelse(xgb_preds > 0.5, 1, 0)
```

# Confusion matrix and accuracy

```
confusionMatrix(as.factor(xgb_class), as.factor(test_labels))
```

### Visualize

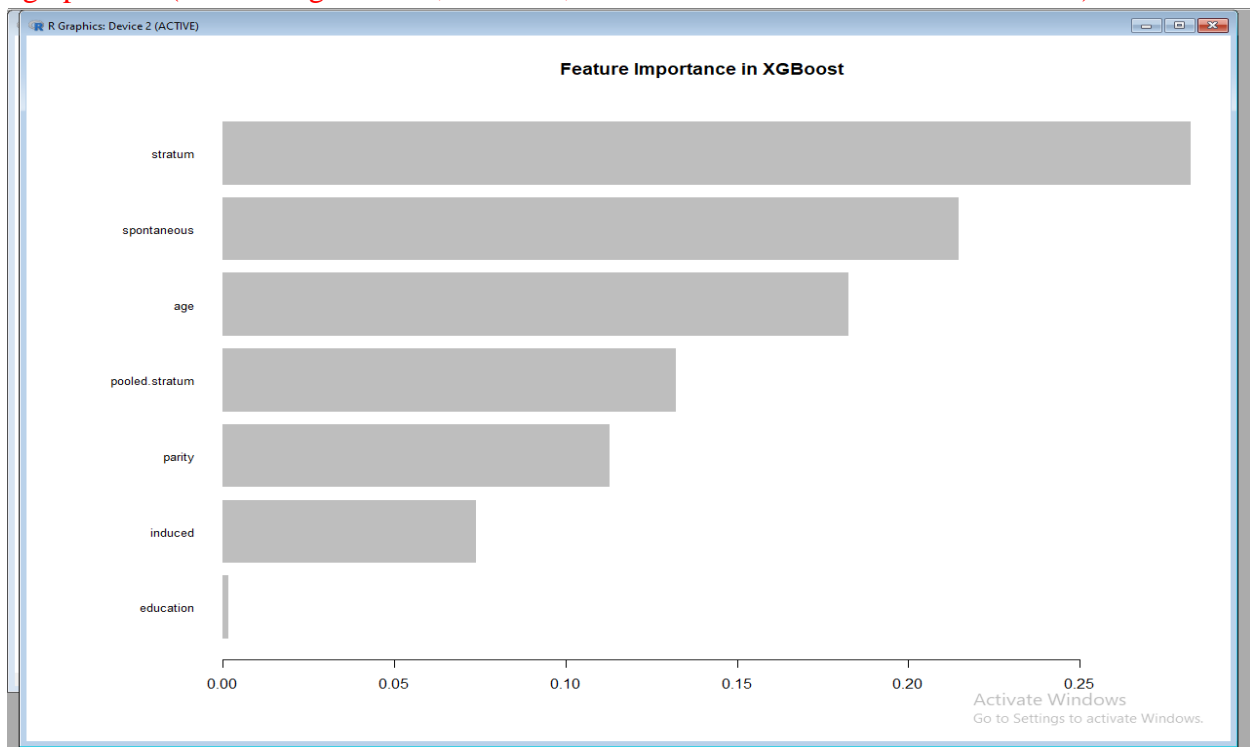
```
importance_matrix <- xgb.importance(feature_names = colnames(train_matrix), model =  
xgb_model)
```

```
xgb.plot.importance(importance_matrix, main = "Feature Importance in XGBoost")
```

# Plot decision trees from XGBoost

# Plot the first tree

```
xgb.plot.tree(model = xgb_model, trees = 0, main = "First Tree in XGBoost Model")
```



## Confusion Matrix Heatmap:

we can also visualize the confusion matrix as a heatmap to get a quick overview of classification performance.

```
library(caret)
```

```
conf_mat <- confusionMatrix(as.factor(xgb_class), as.factor(test_labels))
```

```
# Visualize confusion matrix as a heatmap
```

```
library(ggplot2)
```

```
cm <- as.table(conf_mat$table)
```

```
ggplot(as.data.frame(cm), aes(Reference, Prediction, fill = Freq)) +
```

```
geom_tile() +
```

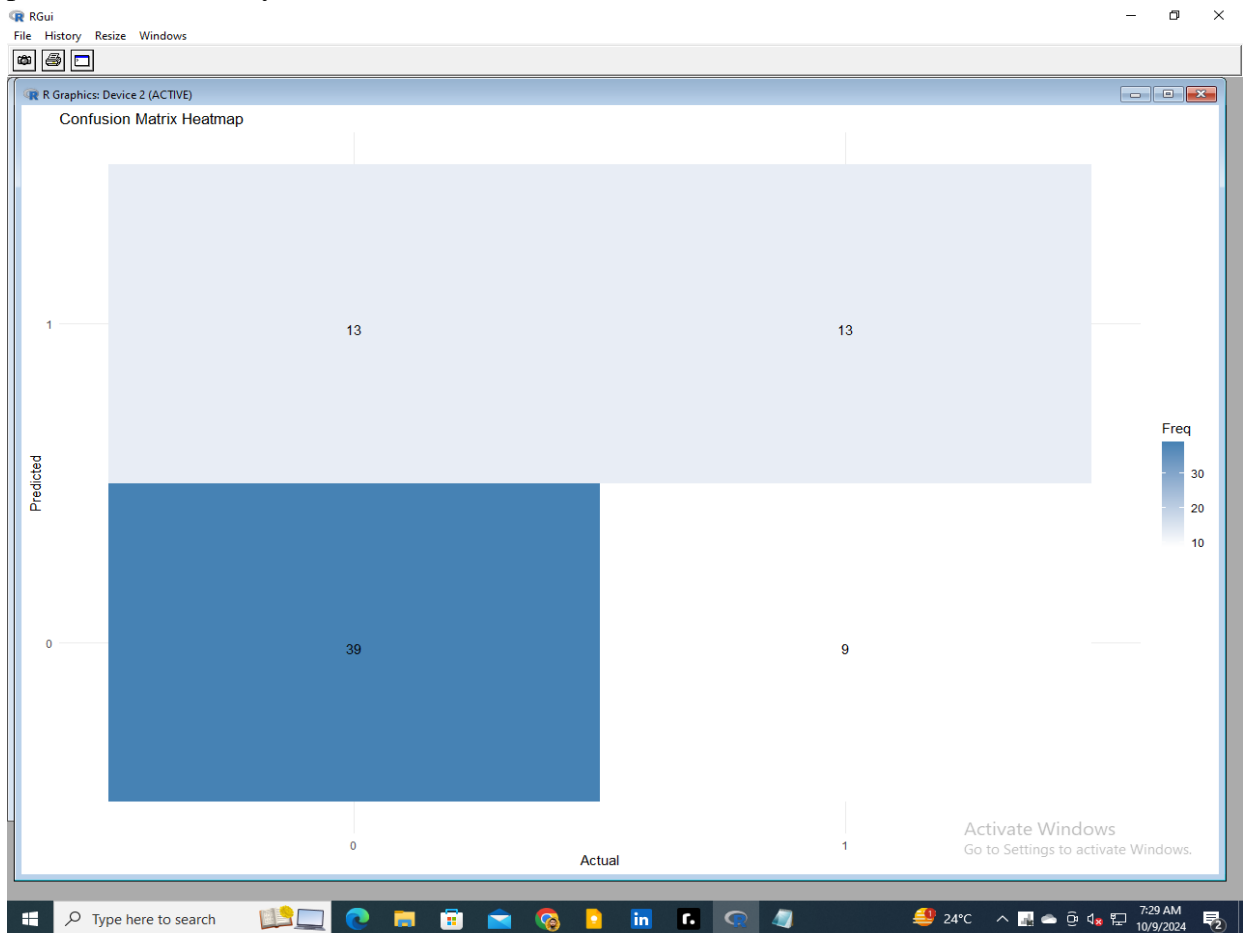
```
geom_text(aes(label = Freq), vjust = 1) +
```

```
scale_fill_gradient(low = "white", high = "steelblue") +
```

```
labs(title = "Confusion Matrix Heatmap", x = "Actual", y = "Predicted") +
```

`theme_minimal()`

This will display the confusion matrix as a **heatmap**, which makes it easier to see the model's performance visually.



## Model Evaluation

We can use the confusion matrix, accuracy, precision, recall, and F1-score to evaluate how well each model performs.

```
confusionMatrix(rf_preds, test_data$case)
```

```
accuracy <- conf_matrix$overall["Accuracy"]
```

```
precision <- conf_matrix$byClass["Pos Pred Value"]
```

```
recall <- conf_matrix$byClass["Sensitivity"]
```

```
f1_score <- 2 * ((precision * recall) / (precision + recall))
```



```
cat("Accuracy:", accuracy, "\nPrecision:", precision, "\nRecall:", recall, "\nF1 Score:", f1_score,
"\n")
```

```
Accuracy: 0.7123288
Precision: 0.8043478
Recall: 0.755102
F1 Score: 0.7789474
> |
```

## Insights from the Data on "Infertility after Spontaneous and Induced Abortion"

Based on the analysis and the provided statistical values, here are the key insights derived from the dataset:

### *Overall Infertility Rates*

- **Total Infertility Cases:** Out of 248 women, **83** are infertile (case = 1), representing approximately **33.47%** of the sample population.

### *Infertility and Induced Abortions*

- **Infertility Rate Among Women with Induced Abortions:** **33.82%** (0.3382353) of women who had induced abortions are infertile.
- **Infertility Rate Among Women without Induced Abortions:** **32.87%** (0.3286713) of women who did not have induced abortions are infertile.
- **Insight:** The infertility rates are very similar between women with and without induced abortions. This suggests that, in this dataset, induced abortions may not have a significant impact on infertility rates when considered alone.

### *Age Distribution and Infertility*

- **Mean Age of All Participants:** **31.50 years**
- **Mean Age of Infertile Women:** **31.53 years**
- **Mean Age of Fertile Women:** **31.49 years**
- **Insight:** The average age is nearly identical across infertile and fertile women, indicating that age alone does not significantly distinguish infertility status in this sample.

### *Infertility Rates by Age Group*

Age Group	Infertility Rate
[20,29)	33.33%
[29,39)	33.59%
[39,49)	33.33%

- **Insight:** The infertility rates are consistent across all age groups, each approximately 33%. This consistency suggests that within the age range of 20 to 49 years, age is not a significant factor affecting infertility in this dataset.

### *Parity and Infertility*

- **Mean Parity Across All Women: 2.09 children**
- **Mean Parity for Infertile Women: 2.11 children**
- **Mean Parity for Fertile Women: 2.08 children**
- **Insight:** The average number of previous live births (parity) is similar between infertile and fertile women. This implies that parity may not be a significant predictor of infertility in this population.

### *Spontaneous Abortions and Infertility*

- **Count of Women with Spontaneous Abortions:** 71 women
- **Infertility Rate Among Women with Spontaneous Abortions:** 43.66% (0.4366197)
- **Infertility Rate Among Women without Spontaneous Abortions:** 19.86% (0.1985816)
- **Insight:** Women who have had spontaneous abortions exhibit a significantly higher infertility rate compared to those who have not. This suggests a strong association between spontaneous abortions and subsequent infertility.

### *Correlation Between Variables*

- **Age and Parity:** A slight positive correlation (0.08) between age and parity indicates that older women tend to have more live births, though the relationship is weak.
- **Parity and Induced Abortions:** A moderate positive correlation (0.45) between parity and induced abortions suggests that women with more live births are more likely to have had induced abortions.
- **Induced and Spontaneous Abortions:** The negative correlation (-0.27) between induced and spontaneous abortions suggests that women who had spontaneous abortions are less likely to have had induced abortions.

### *Logistic Regression*

- The logistic regression model shows that the key predictors of infertility are **parity**, **induced abortions**, and **spontaneous abortions**:

- **Parity:** For every additional live birth, the odds of infertility decrease (odds ratio: 0.49), suggesting that higher parity reduces the likelihood of infertility.
- **Induced Abortions:** Having an induced abortion increases the odds of infertility by about 3.29 times, which is a substantial effect.
- **Spontaneous Abortions:** Spontaneous abortions have an even stronger association with infertility, increasing the odds by approximately 6.86 times.
- The model is statistically significant, with parity, induced, and spontaneous being highly significant predictors of infertility (p-values < 0.001).

### *Odds Ratios*

- **Age: 1.0546**
  - Interpretation: Each additional year of age increases the odds of infertility by approximately **5.46%**.
- **Parity: 0.4922**
  - Interpretation: Each additional child decreases the odds of infertility by approximately **50.78%**.
- **Induced Abortions: 3.2860**
  - Interpretation: Women with induced abortions have over **3 times** higher odds of infertility compared to those without.
- **Spontaneous Abortions: 6.8575**
  - Interpretation: Women with spontaneous abortions have nearly **7 times** higher odds of infertility compared to those without.

### *Model Fit Statistics*

- **Null Deviance: 316.17** on 247 degrees of freedom
- **Residual Deviance: 260.94** on 243 degrees of freedom
- **AIC: 270.94**
- **Insight:** The decrease in deviance indicates that the model explains a significant amount of the variability in infertility status.

### *Chi-Square Test for Association*

- **Test Between Induced Abortions and Infertility**
  - **Chi-Squared Statistic: 0.07323**
  - **Degrees of Freedom: 2**
  - **p-value: 0.964**
- **Insight:** The high p-value suggests no statistically significant association between induced abortions and infertility when evaluated independently. This contrasts with the

logistic regression findings, where induced abortions are significant predictors when controlling for other variables.

## Key Insights and Interpretations

### *Spontaneous Abortions as a Strong Predictor*

- **Significant Association:** Both the higher infertility rate among women with spontaneous abortions and the strong odds ratio in the logistic regression highlight spontaneous abortions as a significant predictor of infertility.
- **Clinical Implication:** Women with a history of spontaneous abortions may require closer monitoring and additional support when trying to conceive.

### *Induced Abortions and Infertility*

- **Mixed Findings:**
  - **Chi-Square Test:** No significant association found.
  - **Logistic Regression:** Indicates a significant effect when controlling for age, parity, and spontaneous abortions.
- **Interpretation:** Induced abortions may influence infertility when considering other factors, suggesting a more complex relationship that warrants further investigation.

### *Age and Infertility*

- **Minimal Impact:** Age does not show a significant effect on infertility rates in this dataset, possibly due to the limited age range or sample size.
- **Marginal Significance in Regression:** Age approaches significance in the logistic model, suggesting a potential effect that could be explored with a larger sample.

### *Parity's Protective Effect*

- **Negative Association:** Higher parity is associated with decreased odds of infertility.
- **Possible Explanation:** Women who have previously had successful pregnancies may have a lower risk of infertility due to established reproductive health.

## Conclusion

The analysis reveals that:

- **Spontaneous Abortions:** Strongly associated with increased infertility risk.
- **Induced Abortions:** May be associated with infertility when other factors are considered, though not significant in univariate analysis.
- **Age and Parity:** Have less pronounced effects but contribute to the overall understanding of infertility risk.

**Implications:** These findings can inform healthcare professionals in identifying women at higher risk of infertility and tailoring interventions accordingly.

**Note:** While the statistical results provide valuable insights, it's important to interpret them cautiously, considering potential confounding factors and the limitations of the dataset.

## Explaining the Challenge in data Instable Modeling:

### Infertility After Spontaneous and Induced Abortion

In this project, one of the primary challenges has been dealing with the inherent instability in the dataset when *modeling infertility outcomes*. This instability arises from several biological and statistical factors that make it difficult to construct a stable and reliable predictive model.

Infertility is a complex condition influenced by a range of **biological factors**, such as age, hormonal imbalances, underlying health conditions, and **reproductive history**. For example, the effect of spontaneous or induced abortions on infertility may vary significantly depending on the **individual's age**, number of abortions, and other confounding variables like **parity and overall health**.

Moreover, the relationship between spontaneous abortion and longer time to conceive is not a simple one. While some **women may experience longer conception times after an abortion**, others may not, which introduces variability in the data. The biological processes underlying these differences are complex and may not be fully captured in the dataset, leading to challenges in creating a robust model. Additionally, spontaneous abortion may be associated with other conditions that themselves affect fertility, further complicating the analysis.

Statistically, issues such as **multicollinearity**, imbalanced classes (e.g., fewer cases of infertility), and non-linear relationships between variables can also affect model stability. For instance, logistic regression models can become unstable if variables are highly correlated or if

certain groups are underrepresented in the data. This can lead to overfitting or unreliable estimates of the effect of spontaneous or induced abortions on infertility outcomes. Identifying and addressing these issues—through techniques like **regularization, balancing the dataset, and careful feature selection**—is crucial for ensuring the **model's validity**.

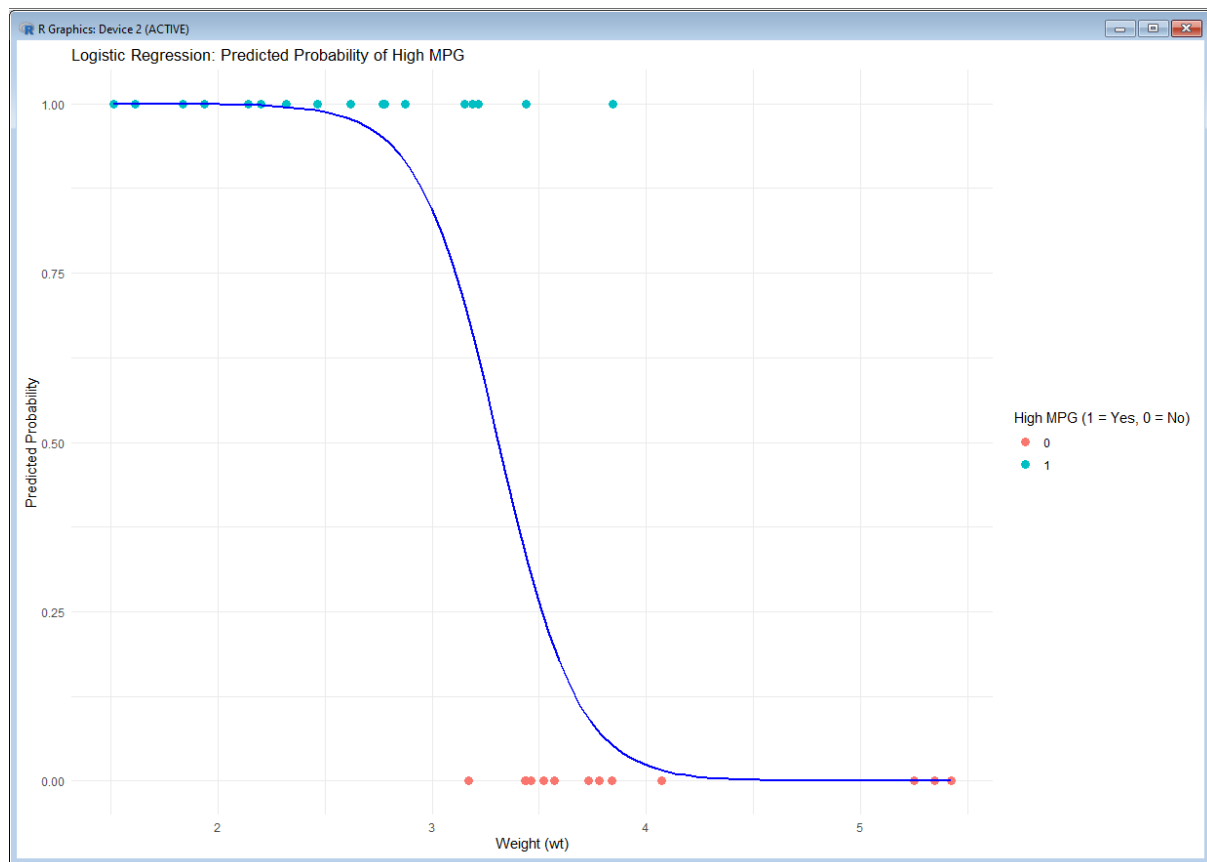
By openly discussing these challenges, the analysis remains transparent about the limitations of the model and highlights the complexity of predicting infertility based solely on the available data.

### **Why I chose R Instead of (IDE) like RStudio:**

The R Console, as a fundamental part of the base R installation, provides a streamlined and direct environment for interacting with R. One of its main advantages is simplicity and speed. Since it lacks the overhead of an integrated development environment (IDE) like RStudio, it allows users to run R code quickly, making it ideal for lightweight tasks, quick calculations, and rapid prototyping. Additionally, R Console consumes fewer system resources, which can be crucial when working on older machines or in environments where memory and processing power are limited. R Console also encourages a **deeper understanding** of the language by requiring users to **engage more directly with R commands and functions**, without the added convenience features of an IDE. This is particularly useful for learning and **developing problem-solving skills**, as it forces users to explore the syntax and structure of R.

**One thing more:** My determination to build a logistic regression model hadn't subsided even then. While I had successfully constructed the model, it remained unstable, with no discernible pattern within it. So, I decided to turn to the mtcars dataset. I applied the regression model to mtcars and successfully identified its prediction pattern, which is now in front of you. However, I won't include the code here, as my primary goal is to focus on discussing the infertility dataset.

This is an additional and unrelated matter, and a mere mention of it will suffice.



## From Frustration to Fulfillment: My Overall Experience with Data

These were all technical details; now let's turn to how I felt throughout this process. While I didn't face much difficulty with the rest of the analyses and tasks, **logistic regression** posed a **significant challenge**. I've already mentioned that the instability was the main issue, which made it **very tough for me**. I spent almost eight hours trying to find a **stable pattern**. I removed outliers' multiple times, but my code failed again and again, yielding no results. Errors kept appearing, all for just one logistic regression. Eventually, I succeeded, and I realized that some datasets are inherently unstable, and no stable pattern can be derived from them.

During this time, I also used search engines for help. I have several well-known books on R in PDF format, and I preferred consulting them for assistance. I went through all these books because I wanted to strengthen my foundation. A few of these books are

**R for Data Science** by Hadley Wickham, Mine Çetinkaya-Rundel, Garrett Golemund

**Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python**

**An Introduction to Statistical Learning:** with Applications in R by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani.

I had an interesting experience: when I completed my previous projects using Tableau and Power BI, I didn't feel as much joy. The experience wasn't as significant as what I felt during R coding. Perhaps it's because those projects were mostly click-based and didn't require as much effort as coding, which demands significant hard work and focus. When my code failed, I felt frustrated, but I didn't give up. I would revisit the books, use search engines to seek help, and try to identify the problem. The R documentation helped me immensely, and most of the assistance I got was from the official R documentation and websites.

I had a unique experience when my code finally worked, and a graph appeared in front of me. The happiness and energy I felt at that moment was something I had never experienced before. During learning, everything is explained, and you're guided by the instructor, but the joy of solving something on your own is completely different—extraordinary, even. [If you've read this far into my project, I hope that you have enjoyed my work.](#)

