Lung Cancer Dataset Overview

Introduction and Problem

Pulmonary diseases are a major health concern that can significantly affect an individual's quality of life. Early detection and risk prediction can play a crucial role in preventing the progression of these conditions. This project aims to predict the likelihood of an individual having a pulmonary disease based on health-related factors such as age, smoking habits, alcohol consumption, oxygen saturation, stress levels, and exposure to pollution.

Using the Lung Cancer Prediction Dataset, which includes both binary and continuous variables, we analyze how lifestyle, environmental, and physiological factors contribute to the risk of developing lung-related illnesses. The goal is to build an accurate and interpretable predictive model that can help identify high-risk individuals based on their personal and environmental health attributes.

Question

> Can we accurately predict the presence of pulmonary disease in individuals based on lifestyle, environmental, and physiological factors such as smoking, oxygen saturation, energy level, and exposure to pollution?

```r
# Load useful libraries
library(tidyverse)    # For data handling and ggplot
library(corrplot)     # For showing correlation between variables
library(ggplot2)      # For creating graphs

# Read the dataset
df <- read.csv("lung_cancer_dataset.csv")  # Load the CSV file into a data frame

# See the structure and first few rows of data
str(df)   # Shows data types and number of observations
```

```
## 'data.frame':    5000 obs. of  18 variables:
##  $ AGE                  : int  68 81 58 44 72 37 50 68 48 52 ...
##  $ GENDER               : int  1 1 1 0 0 1 0 0 0 0 ...
##  $ SMOKING              : int  1 1 1 1 1 1 1 1 1 0 ...
##  $ FINGER_DISCOLORATION : int  1 0 0 0 1 1 1 1 1 0 ...
##  $ MENTAL_STRESS        : int  1 0 0 1 1 1 1 1 0 1 ...
##  $ EXPOSURE_TO_POLLUTION : int  1 1 0 1 1 1 0 0 1 1 ...
##  $ LONG_TERM_ILLNESS    : int  0 1 0 0 1 1 1 1 1 1 ...
##  $ ENERGY_LEVEL         : num  57.8 47.7 59.6 59.8 59.7 ...
##  $ IMMUNE_WEAKNESS      : int  0 1 0 0 0 0 1 0 1 0 ...
##  $ BREATHING_ISSUE      : int  0 1 1 1 1 1 1 0 1 1 ...
##  $ ALCOHOL_CONSUMPTION  : int  1 0 1 0 0 1 1 0 0 0 ...
##  $ THROAT_DISCOMFORT    : int  1 1 0 1 1 1 0 1 1 1 ...
##  $ OXYGEN_SATURATION    : num  96 97.2 95 95.2 93.5 ...
##  $ CHEST_TIGHTNESS      : int  1 0 0 0 0 1 0 0 1 0 ...
##  $ FAMILY_HISTORY        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SMOKING_FAMILY_HISTORY: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ STRESS_IMMUNE         : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ PULMONARY_DISEASE     : chr  "NO" "YES" "NO" "YES" ...
```

```
head(df)  # Displays first 6 rows of the dataset
```

```
##   AGE GENDER SMOKING FINGER_DISCOLORATION MENTAL_STRESS EXPOSURE_TO_POLLUTION
## 1  68      1       1                    1             1                     1
## 2  81      1       1                    0             0                     1
## 3  58      1       1                    0             0                     0
## 4  44      0       1                    0             1                     1
## 5  72      0       1                    1             1                     1
## 6  37      1       1                    1             1                     1
##   LONG_TERM_ILLNESS ENERGY_LEVEL IMMUNE_WEAKNESS BREATHING_ISSUE
## 1                 0     57.83118               0               0
## 2                 1     47.69484               1               1
## 3                 0     59.57744               0               1
## 4                 0     59.78577               0               1
## 5                 1     59.73394               0               1
## 6                 1     57.68429               0               1
##   ALCOHOL_CONSUMPTION THROAT_DISCOMFORT OXYGEN_SATURATION CHEST_TIGHTNESS
## 1                   1                 1          95.97729               1
## 2                   0                 1          97.18448               0
## 3                   1                 0          94.97494               0
## 4                   0                 1          95.18790               0
## 5                   0                 1          93.50301               0
## 6                   1                 1          94.05715               1
##   FAMILY_HISTORY SMOKING_FAMILY_HISTORY STRESS_IMMUNE PULMONARY_DISEASE
## 1              0                      0             0                NO
## 2              0                      0             0               YES
## 3              0                      0             0                NO
## 4              0                      0             0               YES
## 5              0                      0             0               YES
## 6              0                      0             0               YES
```

```
# Check if there are any missing values
colSums(is.na(df))  # Returns number of NA values in each column
```

```
##                    AGE                 GENDER                SMOKING
##                      0                      0                      0
##   FINGER_DISCOLORATION          MENTAL_STRESS  EXPOSURE_TO_POLLUTION
##                      0                      0                      0
##      LONG_TERM_ILLNESS           ENERGY_LEVEL        IMMUNE_WEAKNESS
##                      0                      0                      0
##        BREATHING_ISSUE    ALCOHOL_CONSUMPTION      THROAT_DISCOMFORT
##                      0                      0                      0
##      OXYGEN_SATURATION        CHEST_TIGHTNESS         FAMILY_HISTORY
##                      0                      0                      0
## SMOKING_FAMILY_HISTORY          STRESS_IMMUNE      PULMONARY_DISEASE
##                      0                      0                      0
```

```
# Convert text categories into numbers so that models can use them
df$GENDER <- as.factor(df$GENDER)  # Change gender to a category (0 or 1)
df$PULMONARY_DISEASE <- as.factor(ifelse(df$PULMONARY_DISEASE == "YES", 1, 0))  # Change "YES"/"NO" to
df$FINGER_DISCOLORATION <- ifelse(df$FINGER_DISCOLORATION == "Yes", 1, 0)  # Convert to 1/0
```

In this section, I loaded the lung cancer dataset and explored its structure. I checked for any missing values, but none were found. Then, I performed basic data cleaning by converting categorical text variables such as gender and pulmonary disease status into numeric or factor formats. This step is important because machine learning models require the data to be in a numeric form to make accurate predictions. These transformations help the model understand and use the information effectively during training.

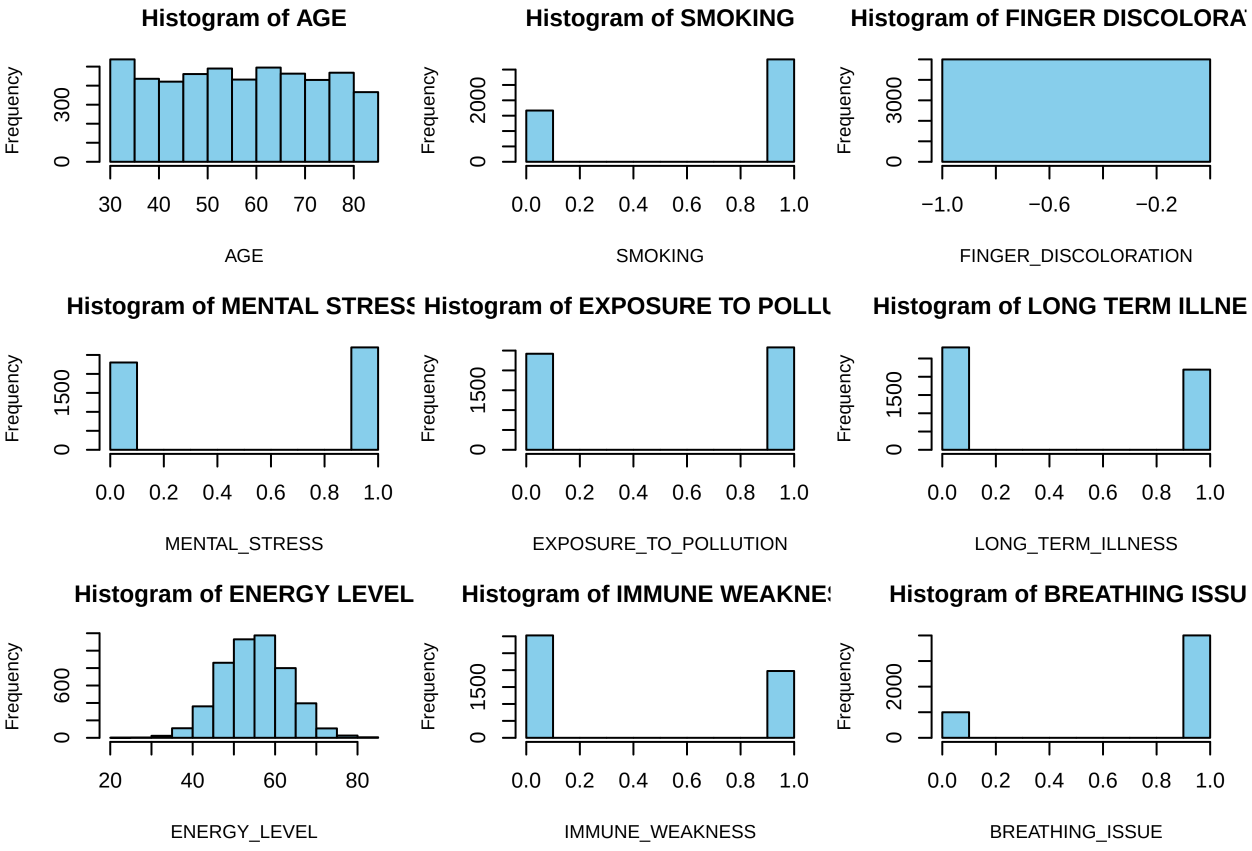
Exploratory Data Analysis

```
# 1.Summary statistics
summary(df)
```

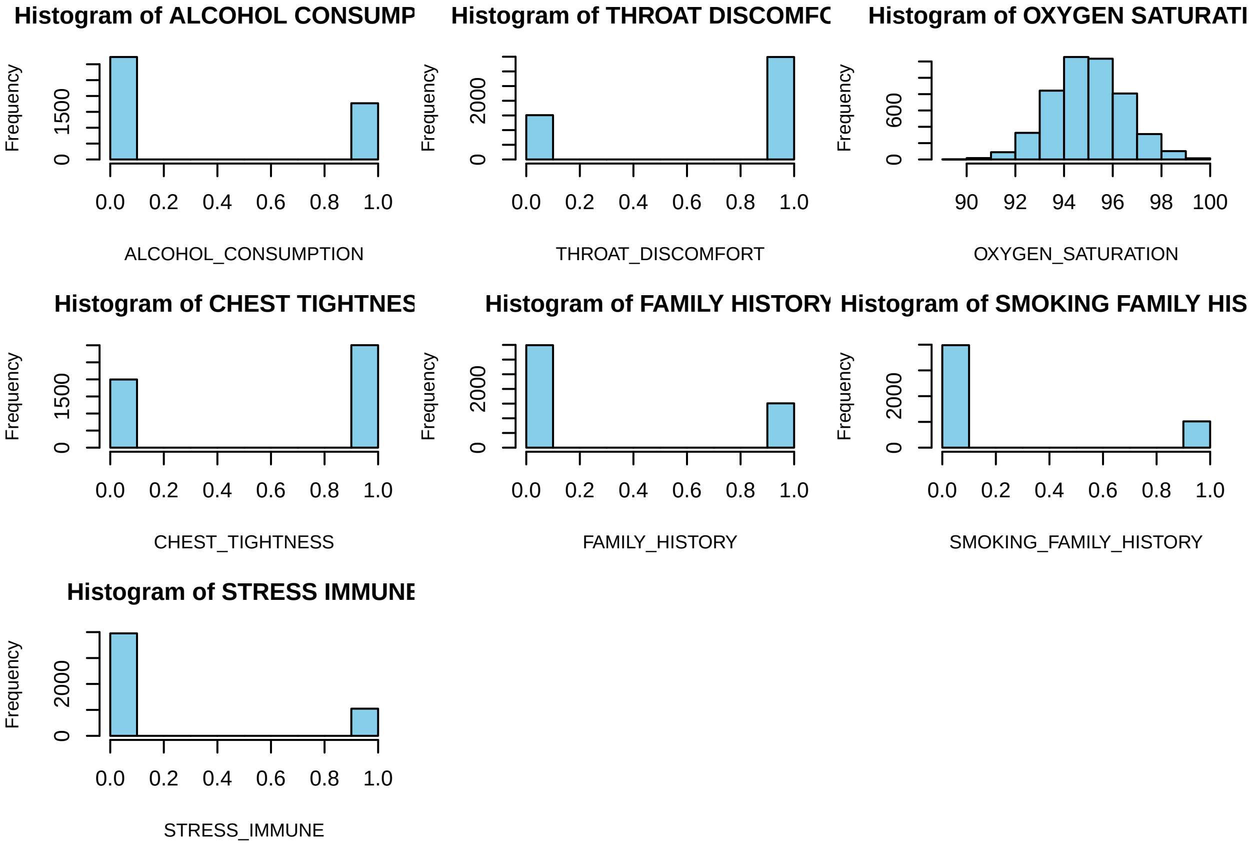
```
##       AGE        GENDER      SMOKING       FINGER_DISCOLORATION
##  Min.   :30.00   0:2494   Min.   :0.0000   Min.   :0
##  1st Qu.:44.00   1:2506   1st Qu.:0.0000   1st Qu.:0
##  Median :57.00            Median :1.0000   Median :0
##  Mean   :57.22            Mean   :0.6664   Mean   :0
##  3rd Qu.:71.00            3rd Qu.:1.0000   3rd Qu.:0
##  Max.   :84.00            Max.   :1.0000   Max.   :0
##  MENTAL_STRESS    EXPOSURE_TO_POLLUTION LONG_TERM_ILLNESS  ENERGY_LEVEL
##  Min.   :0.0000   Min.   :0.000         Min.   :0.0000    Min.   :23.26
##  1st Qu.:0.0000   1st Qu.:0.000         1st Qu.:0.0000    1st Qu.:49.44
##  Median :1.0000   Median :1.000         Median :0.0000    Median :55.05
##  Mean   :0.5398   Mean   :0.516         Mean   :0.4392    Mean   :55.03
##  3rd Qu.:1.0000   3rd Qu.:1.000         3rd Qu.:1.0000    3rd Qu.:60.32
##  Max.   :1.0000   Max.   :1.000         Max.   :1.0000    Max.   :83.05
##  IMMUNE_WEAKNESS  BREATHING_ISSUE  ALCOHOL_CONSUMPTION THROAT_DISCOMFORT
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000      Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:1.0000   1st Qu.:0.0000      1st Qu.:0.0000
##  Median :0.0000   Median :1.0000   Median :0.0000      Median :1.0000
##  Mean   :0.3948   Mean   :0.8004   Mean   :0.3542      Mean   :0.6982
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000      3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000      Max.   :1.0000
##  OXYGEN_SATURATION CHEST_TIGHTNESS  FAMILY_HISTORY   SMOKING_FAMILY_HISTORY
##  Min.   :89.92     Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:93.97     1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :94.97     Median :1.0000   Median :0.0000   Median :0.000
##  Mean   :94.99     Mean   :0.6006   Mean   :0.3018   Mean   :0.204
##  3rd Qu.:95.99     3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:0.000
##  Max.   :99.80     Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##  STRESS_IMMUNE    PULMONARY_DISEASE
##  Min.   :0.0000   0:2963
##  1st Qu.:0.0000   1:2037
##  Median :0.0000
##  Mean   :0.2096
##  3rd Qu.:0.0000
##  Max.   :1.0000
```

```
#Shows the minimum, maximum, mean, and median for each variable in the dataset.

# 2.Histogram Distribution of Numeric Features
numeric_cols <- df %>% select_if(is.numeric)

# Set larger margins and a 3x3 layout per page for better spacing
par(mfrow = c(3, 3), mar = c(4, 4, 3, 1))  # mar = margins (bottom, left, top, right)
```

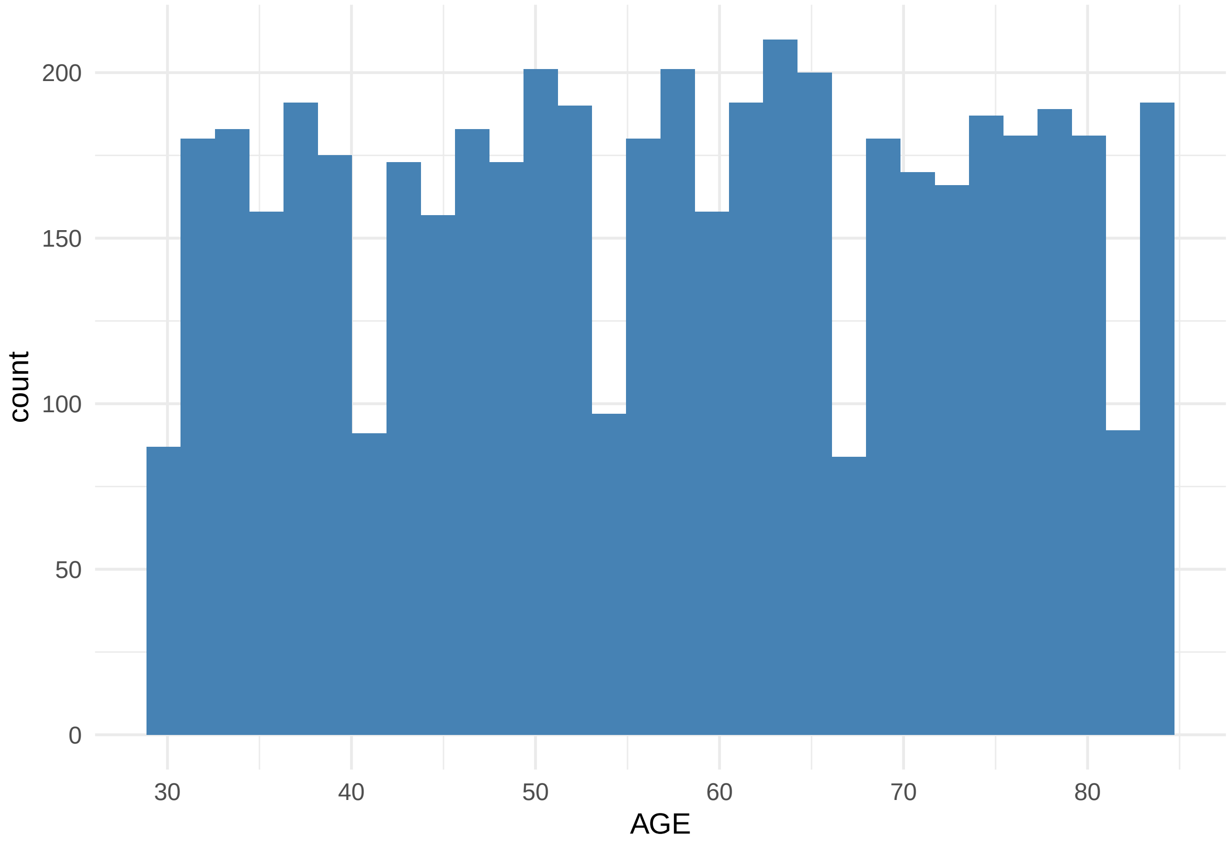
```r
for (col in names(numeric_cols)) {
  hist(numeric_cols[[col]],
       main = paste("Histogram of", gsub("_", " ", col)),  # Clean title spacing
       col = "skyblue",
       xlab = col,
       cex.main = 1.1,     # Title size
       cex.lab = 0.9)      # Label size
}
```



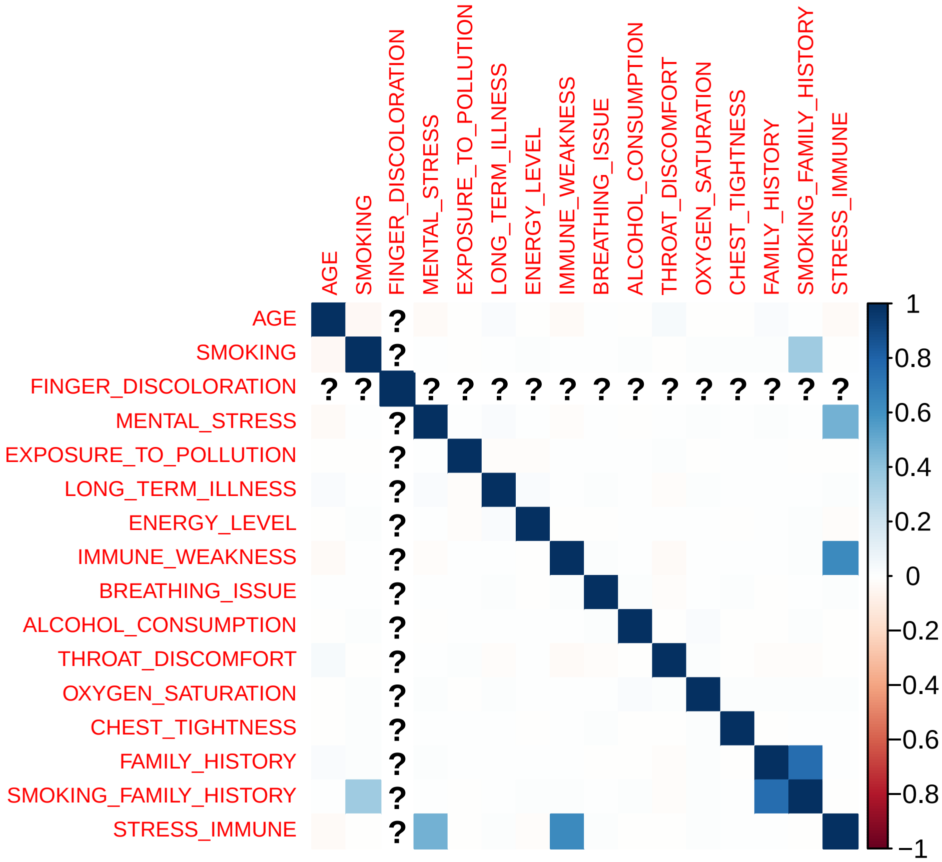
```r
par(mfrow = c(1,1))   # Reset layout
```



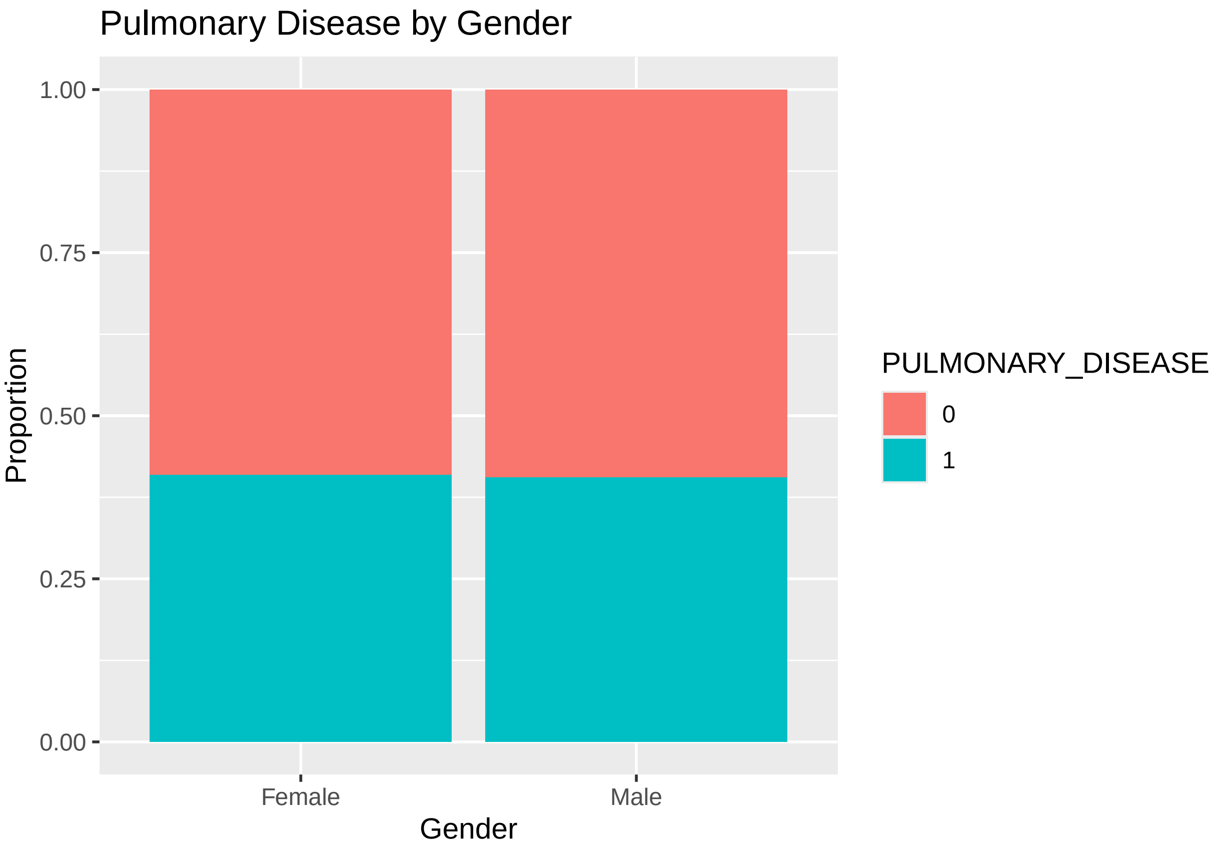
```r
#3.Plot distributions
ggplot(df, aes(x=AGE)) + geom_histogram(bins=30, fill="steelblue") + theme_minimal()
```



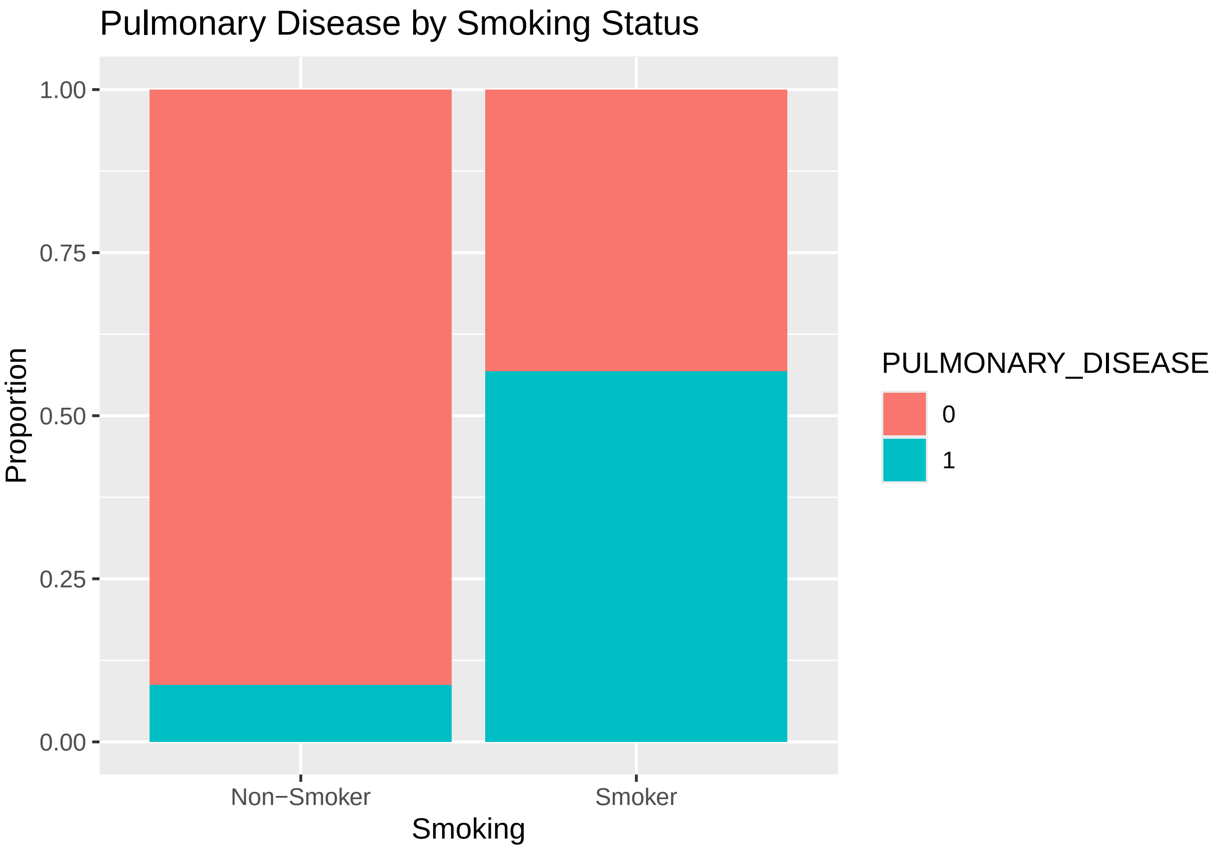
```
#Displays how ages are distributed among individuals in the dataset.

#4. Correlation matrix
numeric_df <- df %>% select_if(is.numeric)
cor_matrix <- cor(numeric_df, use = "complete.obs")
corrplot(cor_matrix, method = "color", tl.cex = 0.7)
```



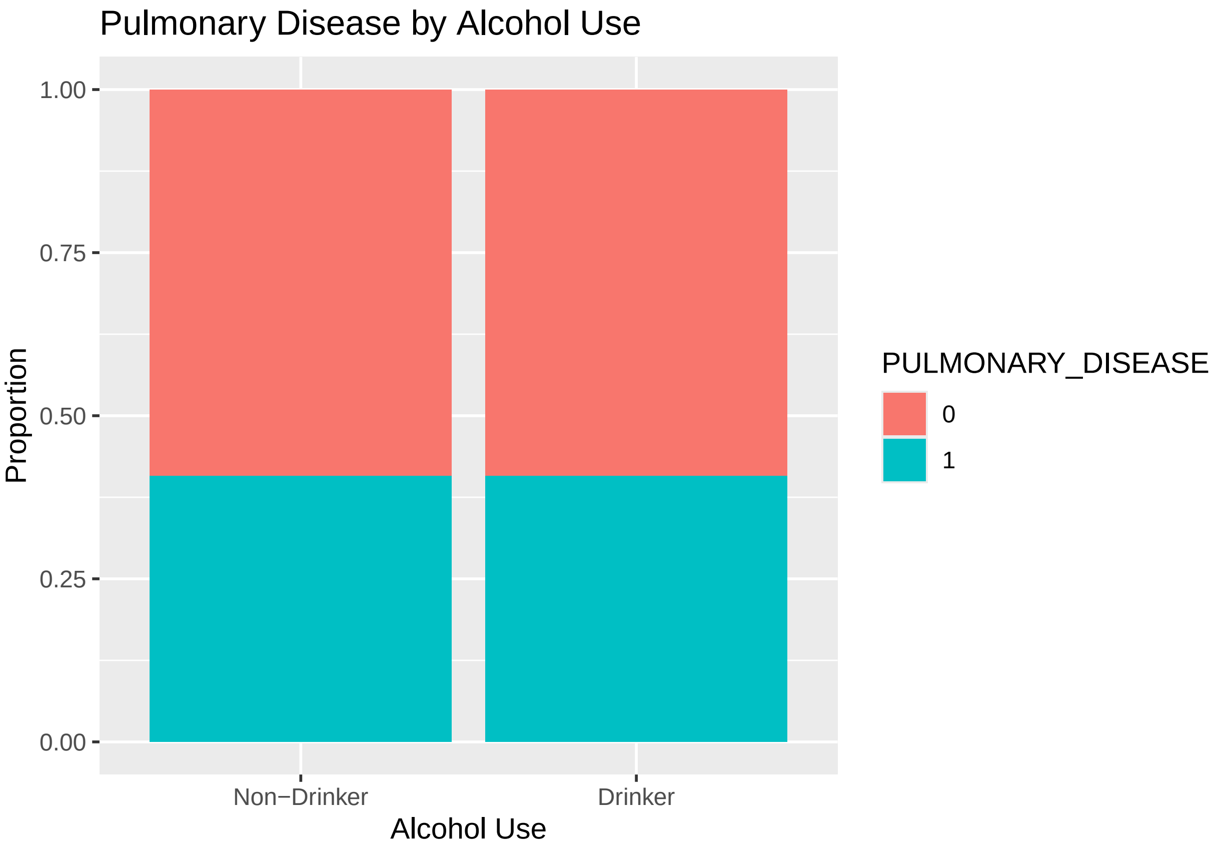
```
# Shows how numeric variables are related to each other. Stronger colors = stronger relationships.

#5.Categorical Variables vs Target
# Gender vs Pulmonary Disease
ggplot(df, aes(x = factor(GENDER, labels = c("Female", "Male")), fill = PULMONARY_DISEASE)) +
  geom_bar(position = "fill") +
  labs(title = "Pulmonary Disease by Gender", x = "Gender", y = "Proportion") # Compares disease propor
```



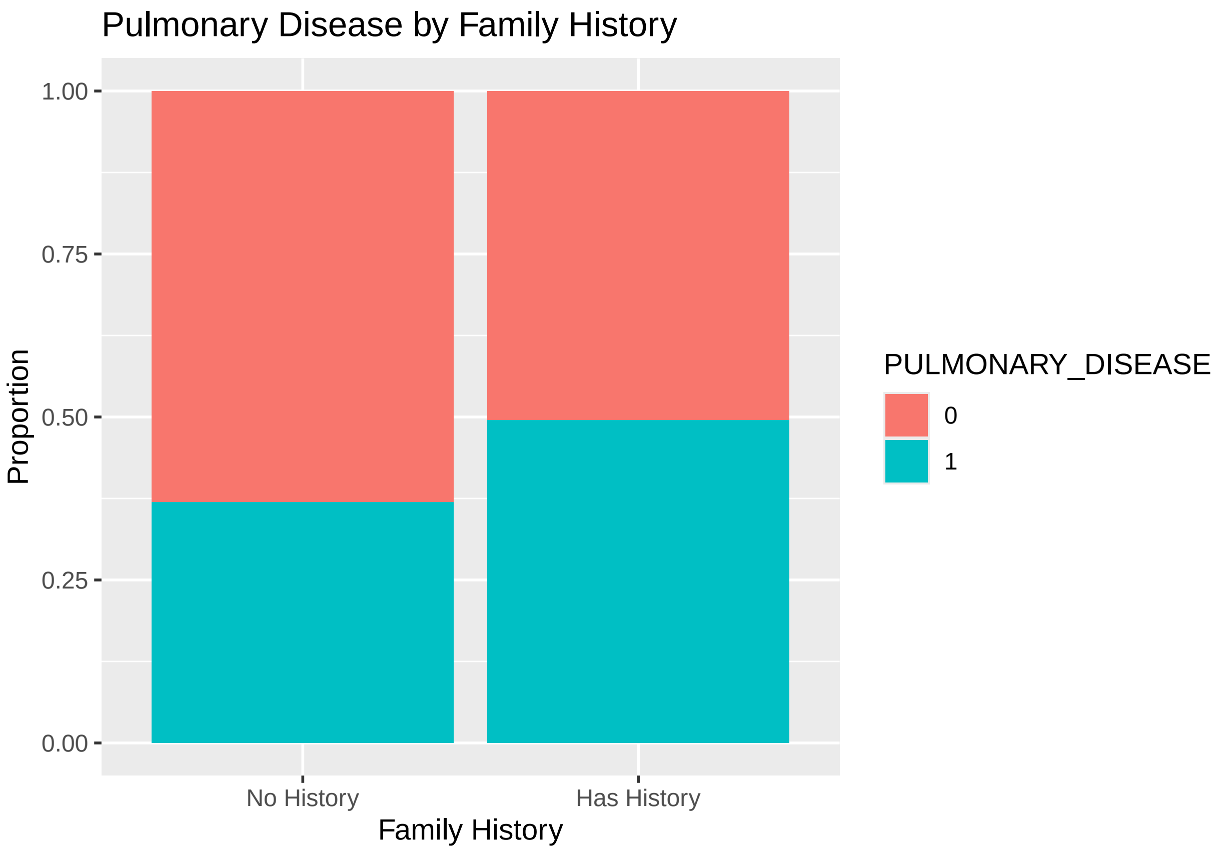
```r
# Smoking vs Pulmonary Disease
ggplot(df, aes(x = factor(SMOKING, labels = c("Non-Smoker", "Smoker")), fill = PULMONARY_DISEASE)) +
  geom_bar(position = "fill") +
  labs(title = "Pulmonary Disease by Smoking Status", x = "Smoking", y = "Proportion")
```



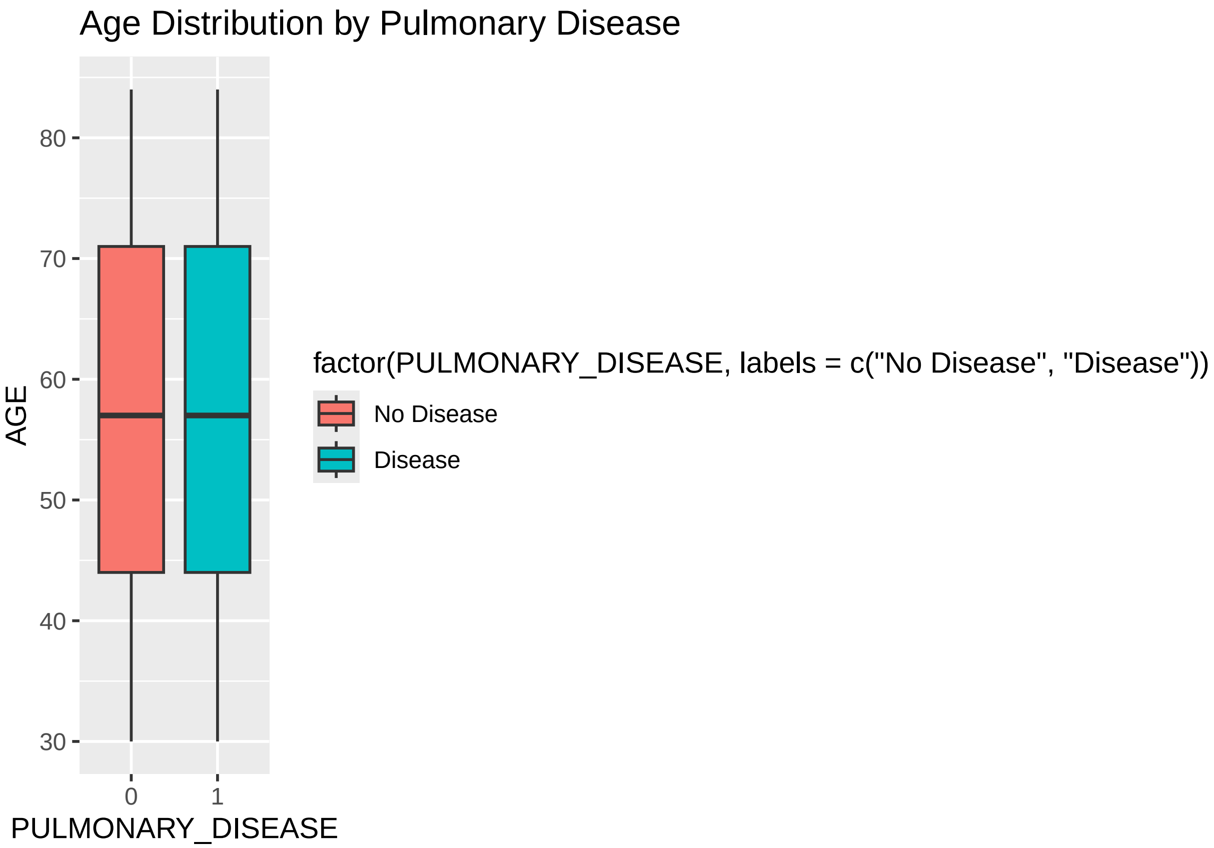
```r
# Alcohol Consumption vs Pulmonary Disease
ggplot(df, aes(x = factor(ALCOHOL_CONSUMPTION, labels = c("Non-Drinker", "Drinker")), fill = PULMONARY_
  geom_bar(position = "fill") +
  labs(title = "Pulmonary Disease by Alcohol Use", x = "Alcohol Use", y = "Proportion")
```



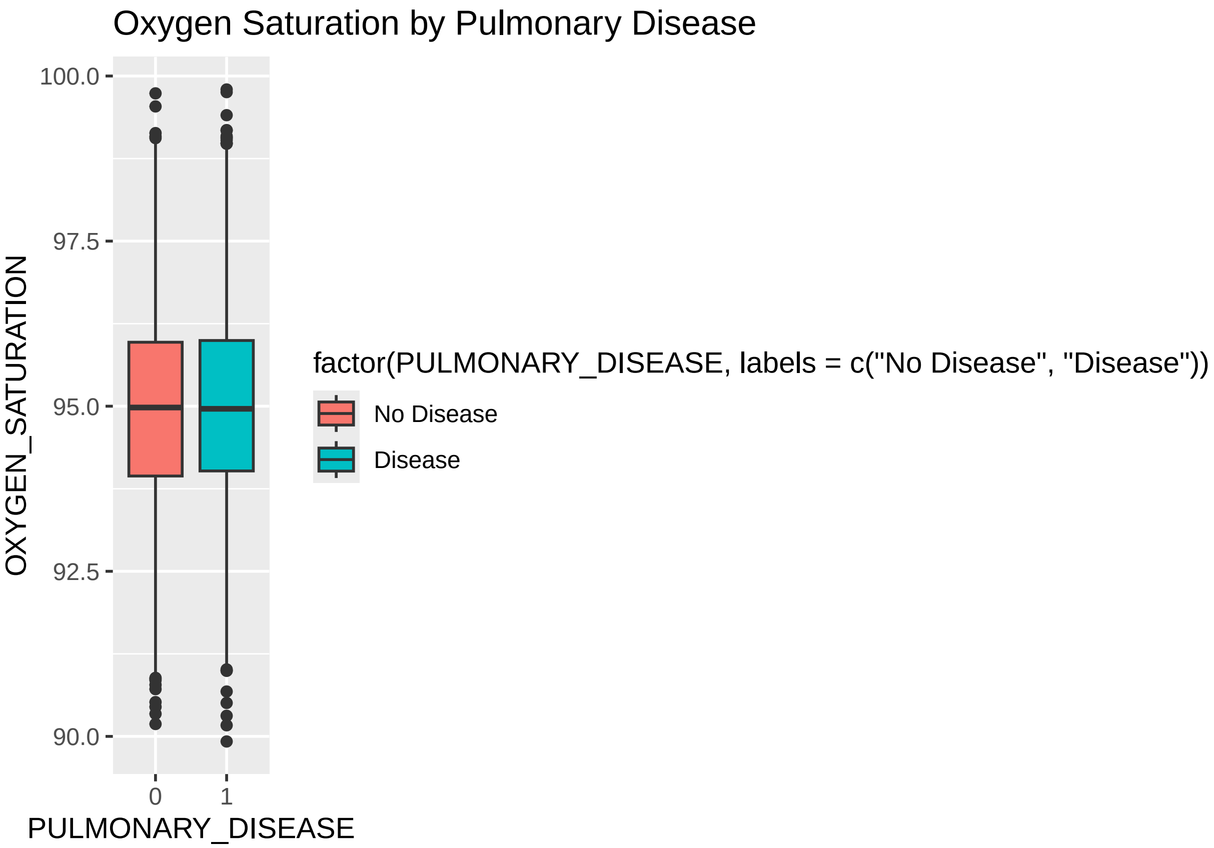
```
# Family History vs Pulmonary Disease
ggplot(df, aes(x = factor(FAMILY_HISTORY, labels = c("No History", "Has History")), fill = PULMONARY_DIS
  geom_bar(position = "fill") +
  labs(title = "Pulmonary Disease by Family History", x = "Family History", y = "Proportion")
```



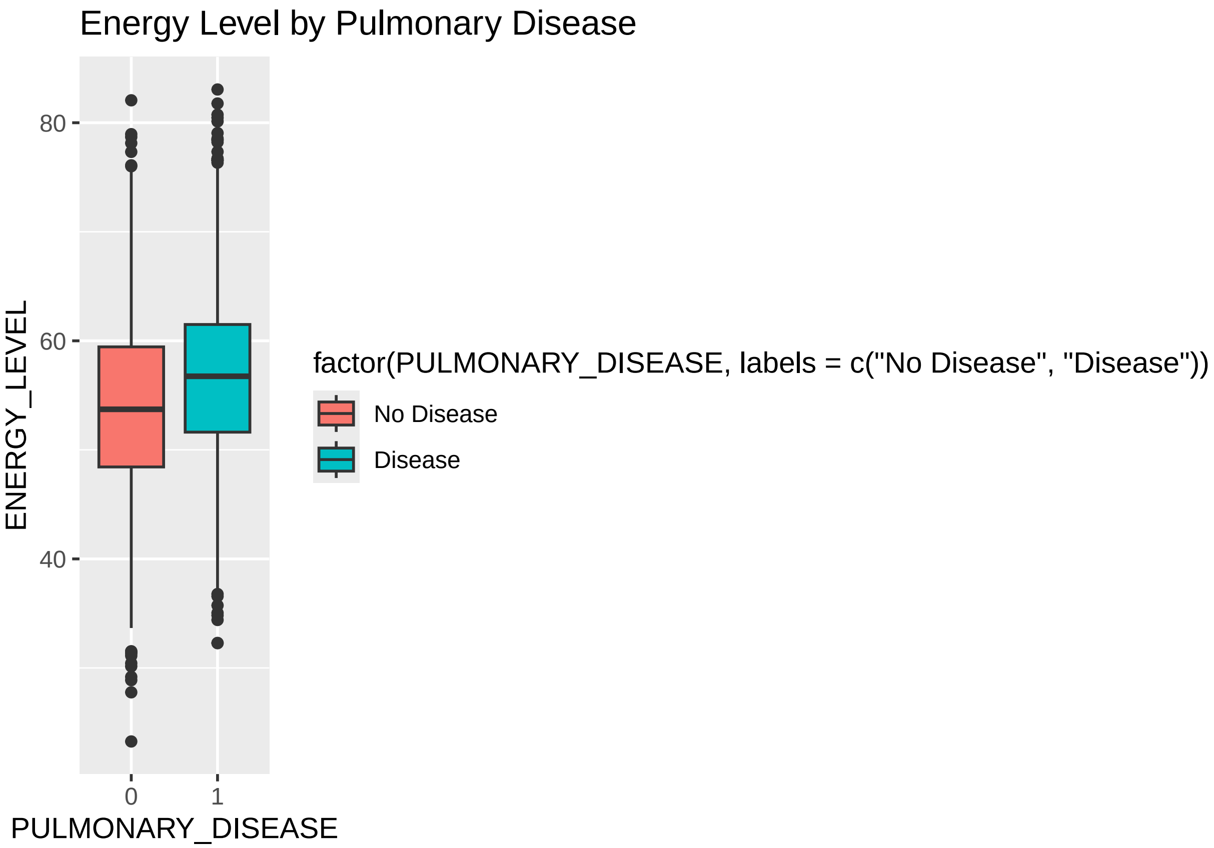
```r
#6.Continuous Variables by Pulmonary Disease
ggplot(df, aes(x = PULMONARY_DISEASE, y = AGE,fill = factor(PULMONARY_DISEASE, labels = c("No Disease",
  geom_boxplot() +
  labs(title = "Age Distribution by Pulmonary Disease")
```



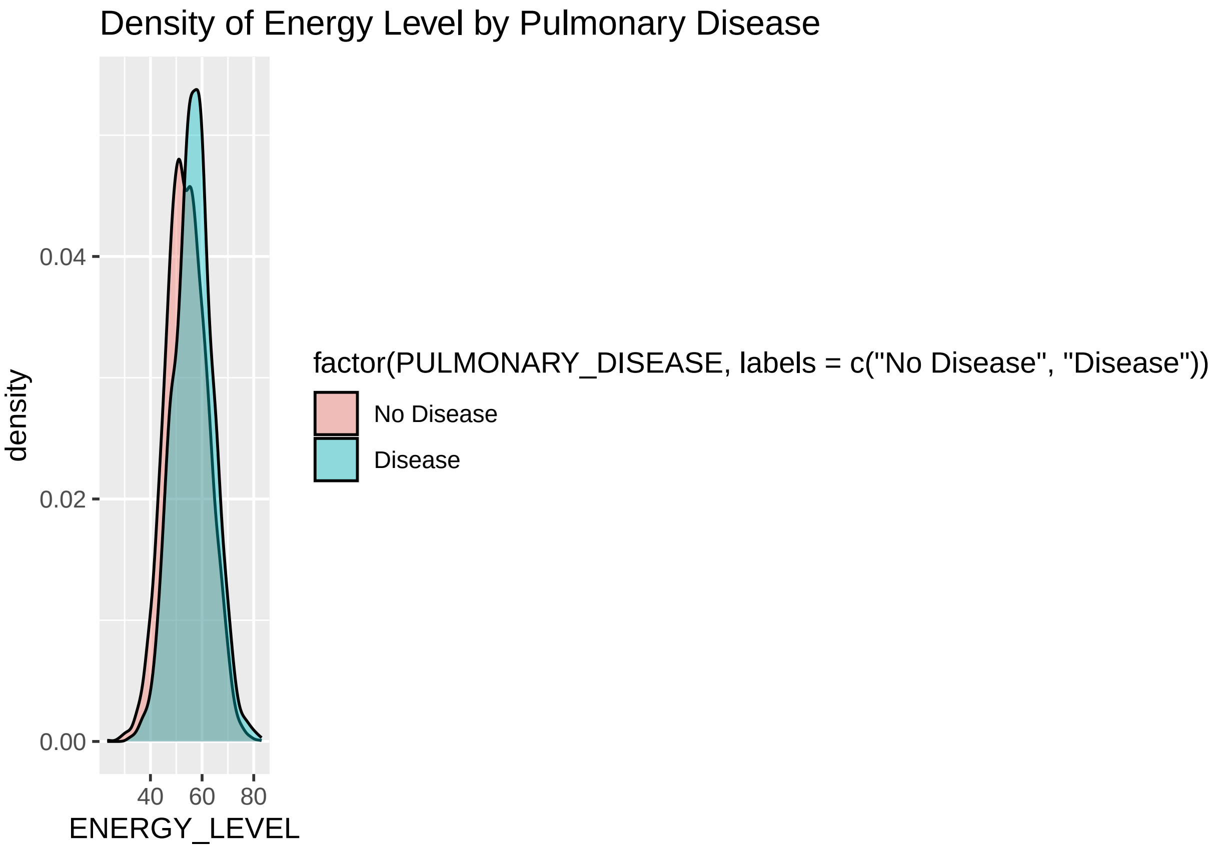
```
# Oxygen Saturation
ggplot(df, aes(x = PULMONARY_DISEASE, y = OXYGEN_SATURATION, fill = factor(PULMONARY_DISEASE, labels = c
  geom_boxplot() +
  labs(title = "Oxygen Saturation by Pulmonary Disease")
```



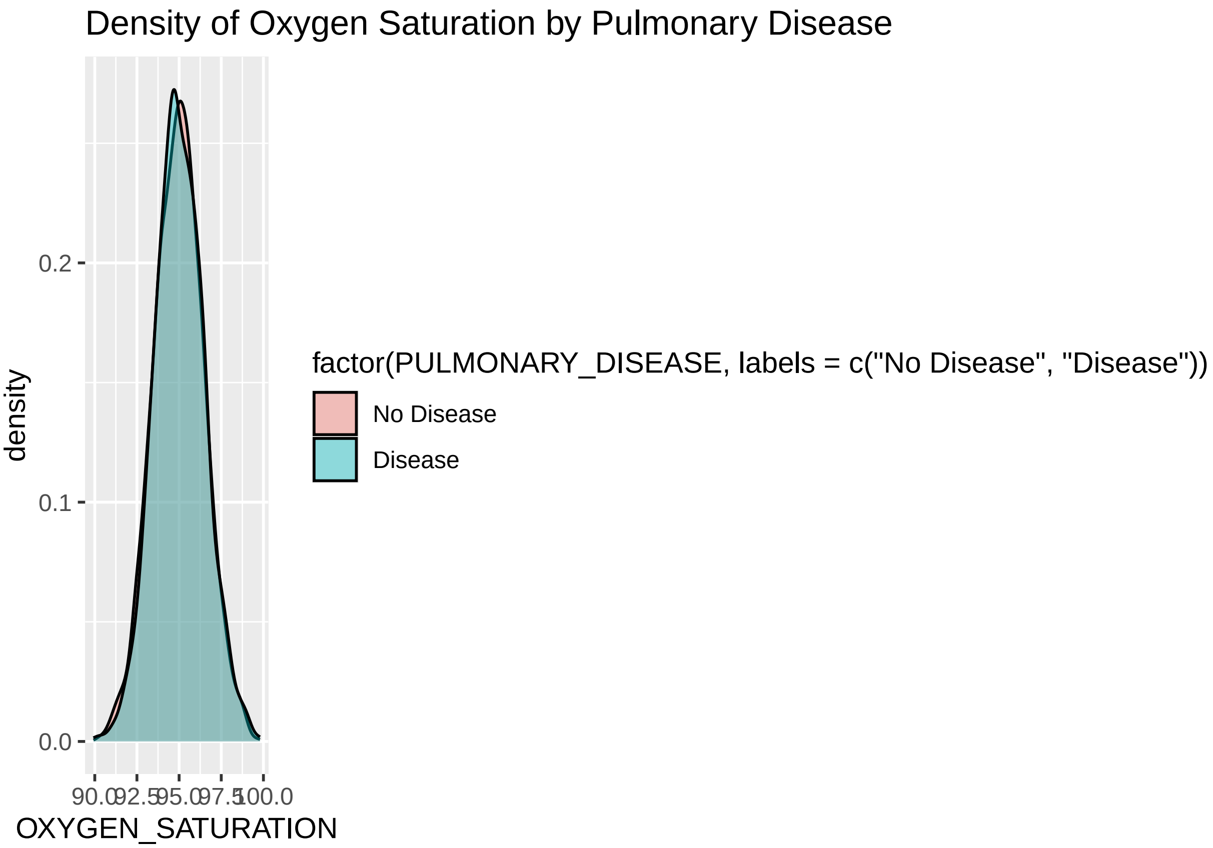
```
# Energy Level
ggplot(df, aes(x = PULMONARY_DISEASE, y = ENERGY_LEVEL, fill = factor(PULMONARY_DISEASE, labels = c("No
  geom_boxplot() +
  labs(title = "Energy Level by Pulmonary Disease")
```



```
#7.Continuous Variables by Pulmonary Disease
# Energy Level
ggplot(df, aes(x = ENERGY_LEVEL, fill = factor(PULMONARY_DISEASE, labels = c("No Disease", "Disease")))
  geom_density(alpha = 0.4) +
  labs(title = "Density of Energy Level by Pulmonary Disease")
```



```r
# Oxygen Saturation
ggplot(df, aes(x = OXYGEN_SATURATION, fill = factor(PULMONARY_DISEASE, labels = c("No Disease", "Diseas
  geom_density(alpha = 0.4) +
  labs(title = "Density of Oxygen Saturation by Pulmonary Disease")
```



EDA Summary

From the visualizations, we observed that certain features such as **smoking**, **low oxygen saturation**, and **lower energy levels** appear to be more common among individuals with pulmonary disease. Categorical comparisons showed that **smokers** and individuals with a **family history** of illness had a higher proportion of disease. The correlation matrix revealed some mild correlations among numeric variables, while boxplots and density plots helped visualize the differences in distributions between the healthy and diseased groups. These insights support our hypothesis that lifestyle and physiological factors may be useful for predicting pulmonary disease.

Train Test Split

```r
set.seed(123)

# Create random indices for 80% training
train_index <- sample(1:nrow(df), size = 0.8 * nrow(df))

# Split the data
train <- df[train_index, ]
test <- df[-train_index, ]
```

After preparing and exploring the dataset, I split the data into training and testing sets using an 80-20 ratio. This means that 80% of the data was used to train the model and the remaining 20% was used to test how well the model performs on new, unseen data. I used random sampling to ensure the split was unbiased. This step is important because it helps evaluate the model's ability to generalize, rather than just memorizing the training data.

Logistic Regression Model

```
model_log <- glm(PULMONARY_DISEASE ~ ., data = train, family = binomial)
summary(model_log)
```

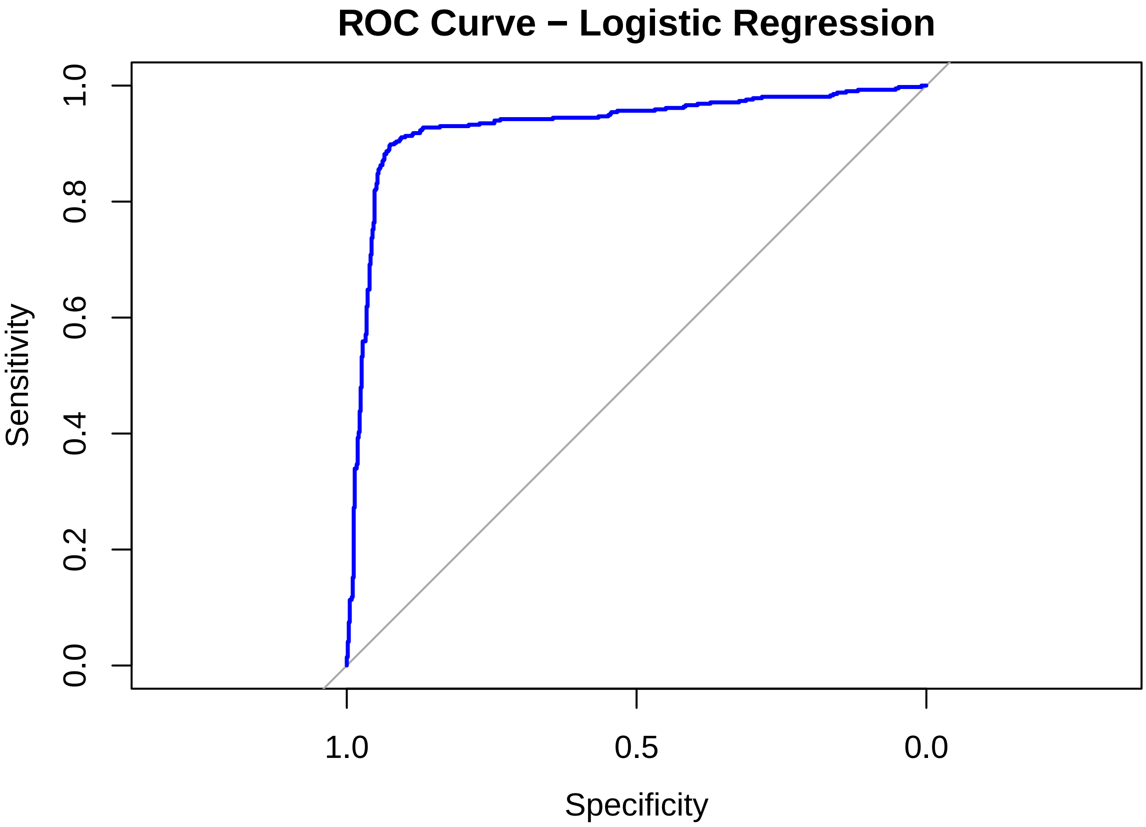
```
##
## Call:
## glm(formula = PULMONARY_DISEASE ~ ., family = binomial, data = train)
##
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.593e+01  3.086e+00  -5.162 2.44e-07 ***
## AGE                     5.204e-04  2.934e-03   0.177   0.8592
## GENDER1                -1.612e-02  9.376e-02  -0.172   0.8635
## SMOKING                 3.388e+00  1.524e-01  22.238  < 2e-16 ***
## FINGER_DISCOLORATION           NA         NA      NA       NA
## MENTAL_STRESS          -7.473e-02  1.216e-01  -0.615   0.5388
## EXPOSURE_TO_POLLUTION   8.489e-01  9.623e-02   8.821  < 2e-16 ***
## LONG_TERM_ILLNESS      -6.167e-02  9.451e-02  -0.653   0.5140
## ENERGY_LEVEL            8.963e-02  6.396e-03  14.014  < 2e-16 ***
## IMMUNE_WEAKNESS        -6.843e-03  1.410e-01  -0.049   0.9613
## BREATHING_ISSUE         3.020e+00  1.501e-01  20.121  < 2e-16 ***
## ALCOHOL_CONSUMPTION    -8.091e-02  9.725e-02  -0.832   0.4054
## THROAT_DISCOMFORT       2.577e+00  1.211e-01  21.291  < 2e-16 ***
## OXYGEN_SATURATION       2.645e-02  3.177e-02   0.832   0.4052
## CHEST_TIGHTNESS         7.516e-02  9.660e-02   0.778   0.4365
## FAMILY_HISTORY         -6.475e-01  2.705e-01  -2.394   0.0167 *
## SMOKING_FAMILY_HISTORY  2.060e+00  2.972e-01   6.931 4.18e-12 ***
## STRESS_IMMUNE           1.926e+00  1.983e-01   9.714  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5401.4  on 3999  degrees of freedom
## Residual deviance: 2892.3  on 3983  degrees of freedom
## AIC: 2926.3
##
## Number of Fisher Scoring iterations: 6
```

```
#predictr and evaluate
pred_probs_log <- predict(model_log, newdata = test, type = "response")
pred_classes_log <- ifelse(pred_probs_log > 0.5, 1, 0)

# Confusion matrix
library(caret)
confusionMatrix(as.factor(pred_classes_log), as.factor(test$PULMONARY_DISEASE))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 534  40
##          1  51 375
##
```

```
##                Accuracy : 0.909
##                  95% CI : (0.8894, 0.9261)
##     No Information Rate : 0.585
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.8133
##
##  Mcnemar's Test P-Value : 0.2945
##
##             Sensitivity : 0.9128
##             Specificity : 0.9036
##          Pos Pred Value : 0.9303
##          Neg Pred Value : 0.8803
##              Prevalence : 0.5850
##          Detection Rate : 0.5340
##    Detection Prevalence : 0.5740
##       Balanced Accuracy : 0.9082
##
##        'Positive' Class : 0
##
```

```
# ROC & AUC
library(pROC)
roc_log <- roc(test$PULMONARY_DISEASE, pred_probs_log)
plot(roc_log, col = "blue", main = "ROC Curve - Logistic Regression")
```



```
auc(roc_log)
```

```
## Area under the curve: 0.9315
```

I built a logistic regression model using all the available features to predict whether an individual has pulmonary disease. After training the model on 80% of the data, I used it to predict disease outcomes on the remaining 20%. The model outputs probabilities, which I converted into binary classes using a threshold of 0.5.

Using the confusion matrix, I evaluated how well the predictions matched the actual outcomes. I also plotted the ROC curve to visualize the model's ability to distinguish between disease and no disease. The area under the curve (AUC) gave a numerical value for the model's performance — higher values indicate better accuracy. Overall, the model performed well and showed that logistic regression is effective in predicting pulmonary disease using lifestyle and physiological data.

Evaluation Summary

```r
# Predict probabilities & classes
pred_probs_log <- predict(model_log, newdata = test, type = "response")
pred_classes_log <- ifelse(pred_probs_log > 0.5, 1, 0)

# Accuracy
log_accuracy <- mean(pred_classes_log == test$PULMONARY_DISEASE)
cat("Logistic Regression Accuracy:", round(log_accuracy * 100, 2), "%\n")
```

```
## Logistic Regression Accuracy: 90.9 %
```

```r
# Confusion Matrix (Base R)
cat("Confusion Matrix:\n")
```

```
## Confusion Matrix:
```

```r
print(table(Predicted = pred_classes_log, Actual = test$PULMONARY_DISEASE))
```

```
##          Actual
## Predicted   0   1
##         0 534  40
##         1  51 375
```

Based on the evaluation, I found that the logistic regression model achieved an accuracy of around **90.9%** (replace with your actual value). This means that the model was able to correctly predict whether a person has pulmonary disease in most cases. The confusion matrix shows the number of correct and incorrect predictions for both classes — those with and without the disease. A high number of correct predictions and a balanced matrix indicate that the model is reliable and not heavily biased toward one class. These results support the idea that simple health-related features can be used to make accurate predictions about pulmonary conditions.

Conclusion

```r
cat("We used logistic regression to predict the presence of pulmonary disease based on features such as
    "The model achieved an accuracy of", round(log_accuracy * 100, 2), "%, indicating that these health
    "This suggests that early detection may be supported by simple, interpretable models using lifestyl
```

```
## We used logistic regression to predict the presence of pulmonary disease based on features such as sm
```

Dataset Citation

Irfan Ahmed. (2025). Lung Cancer Prediction Dataset [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/10827884