Untitled

Edward Hong

2024-12-04

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
train_data <- read.csv('train.csv')
test_data <- read.csv('test.csv')
```

```
summary(train_data)
```

```
##    PatientID           Age            Gender         Ethnicity
##  Min.   :   1.0   Min.   :60.00   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 376.8   1st Qu.:67.00   1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 752.5   Median :75.00   Median :1.0000   Median :0.0000
##  Mean   : 752.5   Mean   :74.91   Mean   :0.5086   Mean   :0.7114
##  3rd Qu.:1128.2   3rd Qu.:83.00   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1504.0   Max.   :90.00   Max.   :1.0000   Max.   :3.0000
##  EducationLevel       BMI           Smoking       AlcoholConsumption
##  Min.   :0.000   Min.   :15.01   Min.   :0.0000   Min.   : 0.002003
##  1st Qu.:1.000   1st Qu.:21.37   1st Qu.:0.0000   1st Qu.: 5.204286
##  Median :1.000   Median :27.76   Median :0.0000   Median : 9.924320
##  Mean   :1.296   Mean   :27.55   Mean   :0.2839   Mean   :10.030205
##  3rd Qu.:2.000   3rd Qu.:33.78   3rd Qu.:1.0000   3rd Qu.:15.140505
##  Max.   :3.000   Max.   :39.93   Max.   :1.0000   Max.   :19.988291
##  PhysicalActivity    DietQuality        SleepQuality    FamilyHistoryAlzheimers
##  Min.   :0.003616   Min.   :0.009385   Min.   : 4.003   Min.   :0.0000
##  1st Qu.:2.538671   1st Qu.:2.302514   1st Qu.: 5.480   1st Qu.:0.0000
##  Median :4.790574   Median :4.979274   Median : 7.100   Median :0.0000
##  Mean   :4.914426   Mean   :4.937305   Mean   : 7.042   Mean   :0.2447
##  3rd Qu.:7.452197   3rd Qu.:7.576618   3rd Qu.: 8.550   3rd Qu.:0.0000
##  Max.   :9.987429   Max.   :9.998346   Max.   :10.000   Max.   :1.0000
##  CardiovascularDisease    Diabetes        Depression       HeadInjury
##  Min.   :0.0000        Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.0000        1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median :0.0000        Median :0.0000   Median :0.0000   Median :0.00000
```

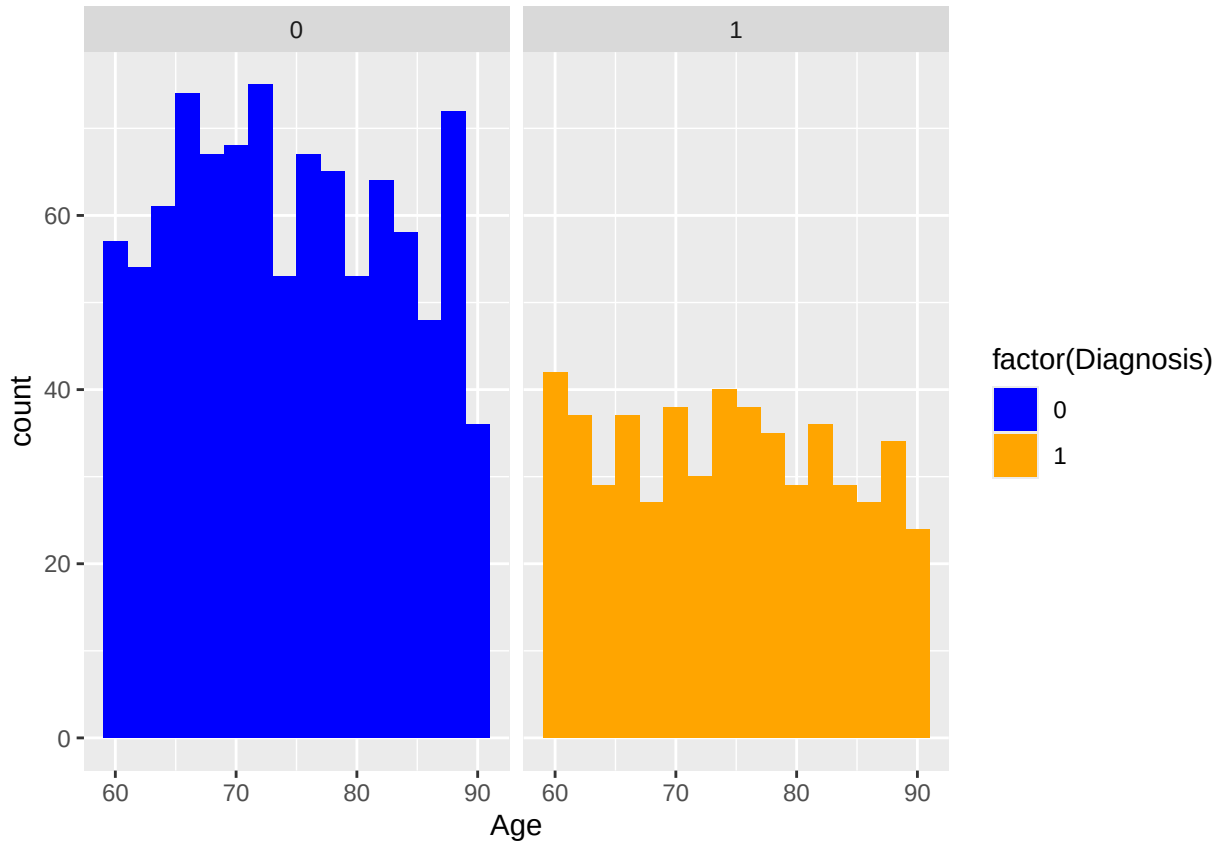
```
##  Mean   :0.1343        Mean   :0.1596   Mean   :0.2081   Mean   :0.09508
##  3rd Qu.:0.0000        3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000
##  Max.   :1.0000        Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
##   Hypertension      SystolicBP     DiastolicBP     CholesterolTotal
##  Min.   :0.0000   Min.   : 90.0   Min.   : 60.00   Min.   :150.1
##  1st Qu.:0.0000   1st Qu.:112.0   1st Qu.: 74.00   1st Qu.:190.5
##  Median :0.0000   Median :135.0   Median : 90.00   Median :224.4
##  Mean   :0.1516   Mean   :134.7   Mean   : 89.71   Mean   :225.2
##  3rd Qu.:0.0000   3rd Qu.:156.0   3rd Qu.:105.00   3rd Qu.:262.5
##  Max.   :1.0000   Max.   :179.0   Max.   :119.00   Max.   :300.0
##  CholesterolLDL   CholesterolHDL  CholesterolTriglycerides      MMSE
##  Min.   : 50.40   Min.   :20.00   Min.   : 50.41           Min.   : 0.0353
##  1st Qu.: 87.52   1st Qu.:39.15   1st Qu.:136.31           1st Qu.: 7.1155
##  Median :124.52   Median :59.59   Median :229.55           Median :14.3225
##  Mean   :124.88   Mean   :59.51   Mean   :226.90           Mean   :14.6491
##  3rd Qu.:161.96   3rd Qu.:78.91   3rd Qu.:313.06           3rd Qu.:21.8386
##  Max.   :199.97   Max.   :99.98   Max.   :399.94           Max.   :29.9914
##  FunctionalAssessment MemoryComplaints BehavioralProblems      ADL
##  Min.   :0.00046      Min.   :0.0000   Min.   :0.0000     Min.   :0.004354
##  1st Qu.:2.65883      1st Qu.:0.0000   1st Qu.:0.0000     1st Qu.:2.358590
##  Median :5.19113      Median :0.0000   Median :0.0000     Median :4.877862
##  Mean   :5.13989      Mean   :0.2055   Mean   :0.1516     Mean   :4.903536
##  3rd Qu.:7.61636      3rd Qu.:0.0000   3rd Qu.:0.0000     3rd Qu.:7.517219
##  Max.   :9.99647      Max.   :1.0000   Max.   :1.0000     Max.   :9.972663
##    Confusion      Disorientation   PersonalityChanges DifficultyCompletingTasks
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000     Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000     1st Qu.:0.0000
##  Median :0.0000   Median :0.0000   Median :0.0000     Median :0.0000
##  Mean   :0.2028   Mean   :0.1562   Mean   :0.1569     Mean   :0.1622
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000     3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000     Max.   :1.0000
##  Forgetfulness      Diagnosis      DoctorInCharge
##  Min.   :0.0000   Min.   :0.0000   Length:1504
##  1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :0.2999   Mean   :0.3537
##  3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000
```

```
summary(test_data)
```

```
##    PatientID         Age            Gender         Ethnicity
##  Min.   :1505   Min.   :60.00   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1666   1st Qu.:67.00   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :1827   Median :75.00   Median :1.0000   Median :0.0000
##  Mean   :1827   Mean   :74.92   Mean   :0.5008   Mean   :0.6651
##  3rd Qu.:1988   3rd Qu.:83.00   3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :2149   Max.   :90.00   Max.   :1.0000   Max.   :3.0000
##  EducationLevel       BMI           Smoking       AlcoholConsumption
##  Min.   :0.000   Min.   :15.01   Min.   :0.0000   Min.   : 0.0105
##  1st Qu.:1.000   1st Qu.:21.98   1st Qu.:0.0000   1st Qu.: 5.0973
##  Median :1.000   Median :27.98   Median :0.0000   Median : 9.9835
##  Mean   :1.265   Mean   :27.90   Mean   :0.2992   Mean   :10.0610
##  3rd Qu.:2.000   3rd Qu.:34.10   3rd Qu.:1.0000   3rd Qu.:15.2282
##  Max.   :3.000   Max.   :39.99   Max.   :1.0000   Max.   :19.9893
```

```
##  PhysicalActivity   DietQuality       SleepQuality   FamilyHistoryAlzheimers
##  Min.   :0.06549   Min.   :0.01645   Min.   :4.004   Min.   :0.0000
##  1st Qu.:2.71470   1st Qu.:2.88894   1st Qu.:5.532   1st Qu.:0.0000
##  Median :4.63731   Median :5.29123   Median :7.164   Median :0.0000
##  Mean   :4.93367   Mean   :5.12333   Mean   :7.073   Mean   :0.2698
##  3rd Qu.:7.39254   3rd Qu.:7.51970   3rd Qu.:8.647   3rd Qu.:1.0000
##  Max.   :9.97658   Max.   :9.99720   Max.   :9.989   Max.   :1.0000
##  CardiovascularDisease    Diabetes        Depression       HeadInjury
##  Min.   :0.0000        Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.0000        1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median :0.0000        Median :0.0000   Median :0.0000   Median :0.00000
##  Mean   :0.1674        Mean   :0.1302   Mean   :0.1829   Mean   :0.08682
##  3rd Qu.:0.0000        3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000
##  Max.   :1.0000        Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
##   Hypertension      SystolicBP     DiastolicBP     CholesterolTotal
##  Min.   :0.0000   Min.   : 90.0   Min.   : 60.00   Min.   :150.1
##  1st Qu.:0.0000   1st Qu.:110.0   1st Qu.: 75.00   1st Qu.:190.0
##  Median :0.0000   Median :131.0   Median : 92.00   Median :226.2
##  Mean   :0.1426   Mean   :133.2   Mean   : 90.18   Mean   :225.3
##  3rd Qu.:0.0000   3rd Qu.:157.0   3rd Qu.:105.00   3rd Qu.:261.1
##  Max.   :1.0000   Max.   :179.0   Max.   :119.00   Max.   :300.0
##  CholesterolLDL   CholesterolHDL  CholesterolTriglycerides      MMSE
##  Min.   : 50.23   Min.   :20.26   Min.   : 50.46           Min.   : 0.005312
##  1st Qu.: 86.03   1st Qu.:38.68   1st Qu.:144.18           1st Qu.: 7.305482
##  Median :120.44   Median :59.81   Median :235.77           Median :14.656169
##  Mean   :123.08   Mean   :59.34   Mean   :231.50           Mean   :15.002261
##  3rd Qu.:160.48   3rd Qu.:79.06   3rd Qu.:316.56           3rd Qu.:22.775011
##  Max.   :199.46   Max.   :99.81   Max.   :399.73           Max.   :29.959425
##  FunctionalAssessment MemoryComplaints BehavioralProblems      ADL
##  Min.   :0.01519      Min.   :0.000    Min.   :0.000      Min.   : 0.001288
##  1st Qu.:2.41528      1st Qu.:0.000    1st Qu.:0.000      1st Qu.: 2.273081
##  Median :4.97614      Median :0.000    Median :0.000      Median : 5.492566
##  Mean   :4.94054      Mean   :0.214    Mean   :0.169      Mean   : 5.168154
##  3rd Qu.:7.39108      3rd Qu.:0.000    3rd Qu.:0.000      3rd Qu.: 7.765987
##  Max.   :9.92794      Max.   :1.000    Max.   :1.000      Max.   : 9.999747
##    Confusion      Disorientation   PersonalityChanges DifficultyCompletingTasks
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000     Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000     1st Qu.:0.0000
##  Median :0.0000   Median :0.0000   Median :0.0000     Median :0.0000
##  Mean   :0.2109   Mean   :0.1628   Mean   :0.1364     Mean   :0.1504
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000     3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000     Max.   :1.0000
##  Forgetfulness    DoctorInCharge
##  Min.   :0.0000   Length:645
##  1st Qu.:0.0000   Class :character
##  Median :0.0000   Mode  :character
##  Mean   :0.3054
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

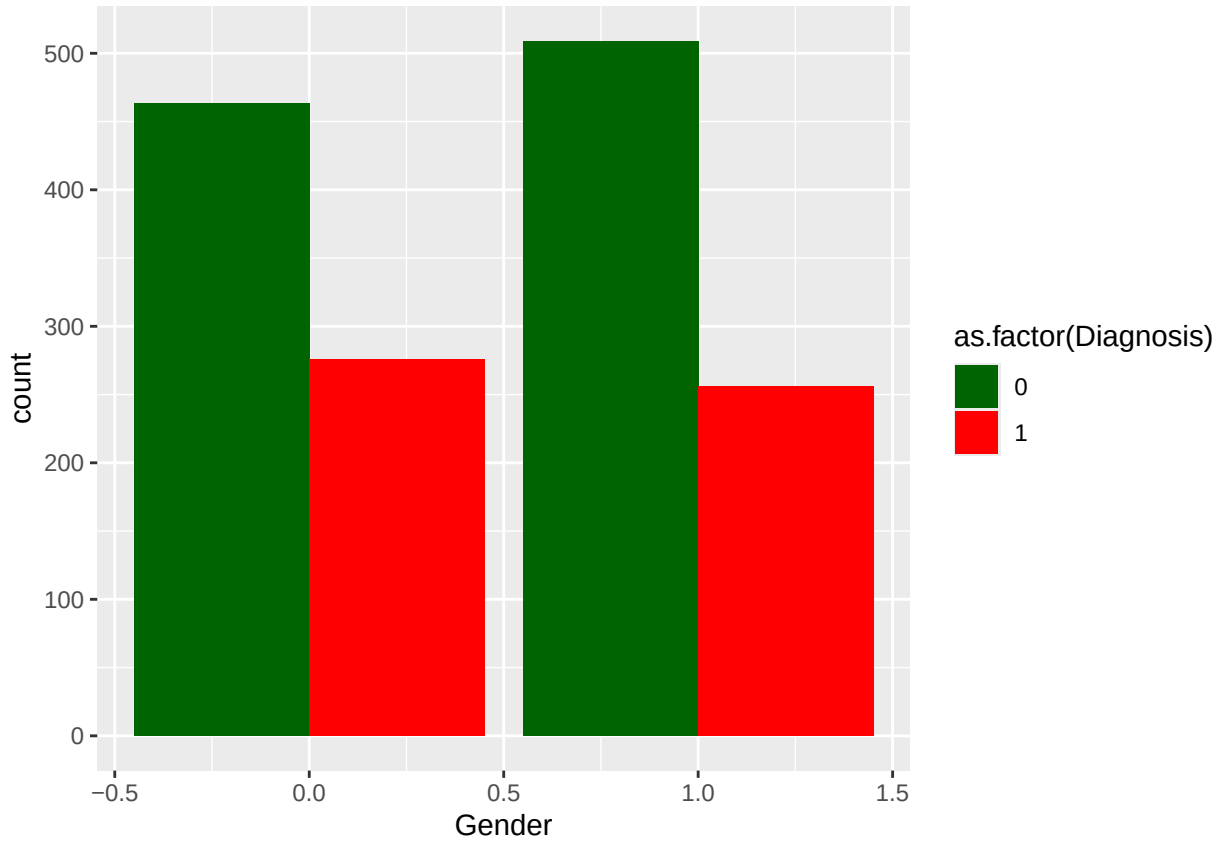
```r
ggplot(train_data, aes(x = Age, fill = factor(Diagnosis))) +
  geom_histogram(binwidth = 2) +
  facet_wrap(~ Diagnosis) +
  scale_fill_manual(values = c("blue", "orange"))
```




```
  labs(title = "Age Distribution by Alzheimer's Diagnosis",
       x = "Age",
       fill = "Diagnosis") +
  theme_minimal()
```

```
## NULL
```

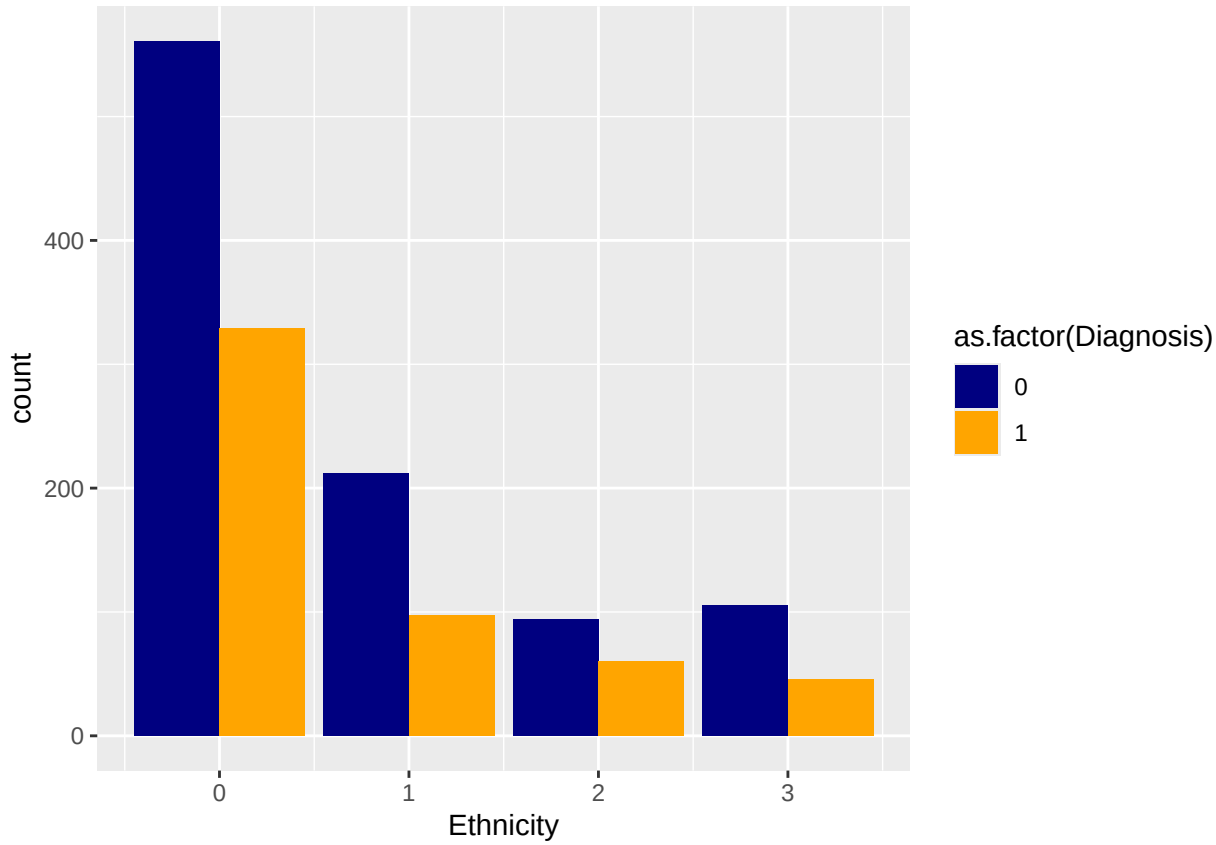
```
ggplot(train_data, aes(x = Gender, fill = as.factor(Diagnosis))) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = c("darkgreen", "red"))
```




```
  labs(title = "Gender Breakdown by Diagnosis",
       x = "Gender",
       y = "Count",
       fill = "Diagnosis") +
  theme_minimal()
```

```
## NULL
```

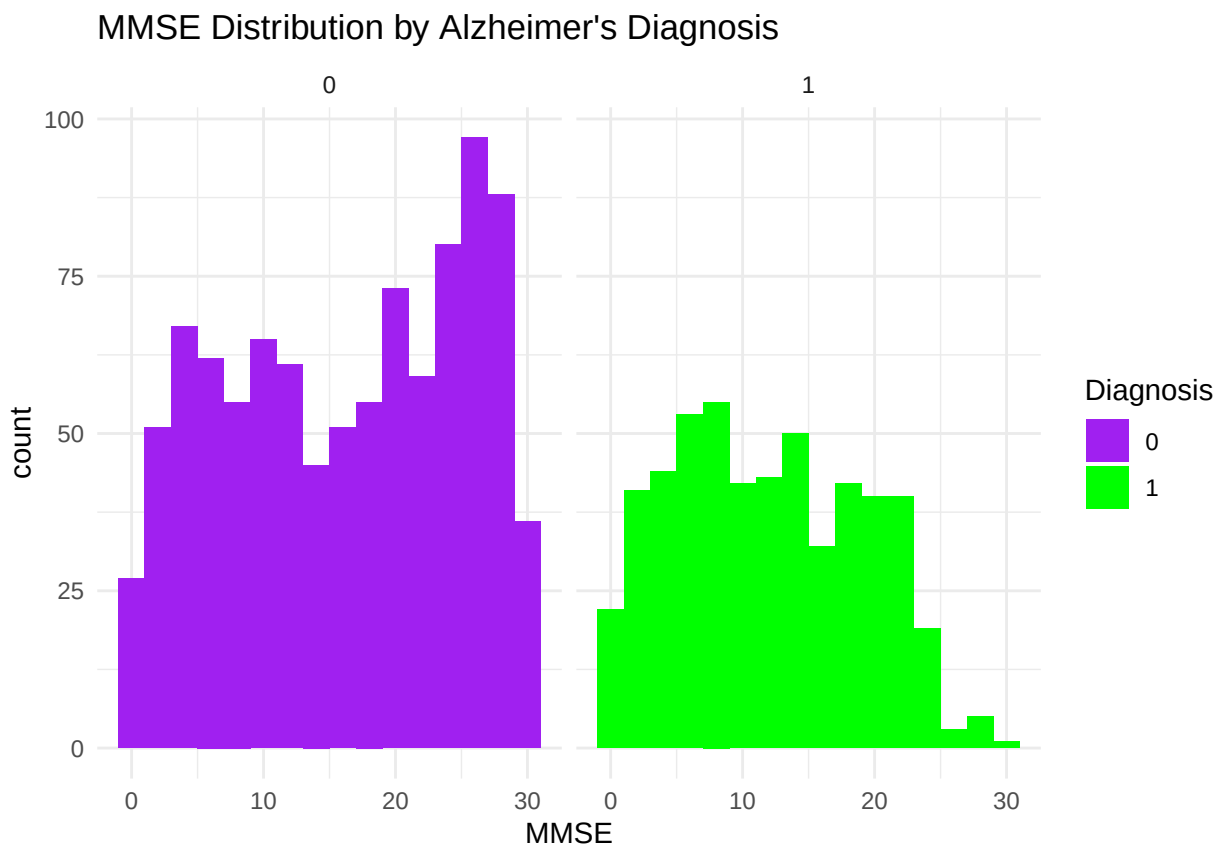
```
ggplot(train_data, aes(x = Ethnicity, fill = as.factor(Diagnosis))) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = c("navy", "orange"))
```




```
  labs(title = "Ethnicity Breakdown by Diagnosis",
       x = "Ethnicity",
       y = "Count",
       fill = "Diagnosis") +
  theme_minimal()
```

```
## NULL
```

```
ggplot(train_data, aes(x = MMSE, fill = factor(Diagnosis))) +
  geom_histogram(binwidth = 2) +
  facet_wrap(~ Diagnosis) +
  scale_fill_manual(values = c("purple", "green")) +
  labs(title = "MMSE Distribution by Alzheimer's Diagnosis",
       x = "MMSE",
       fill = "Diagnosis") +
  theme_minimal()
```



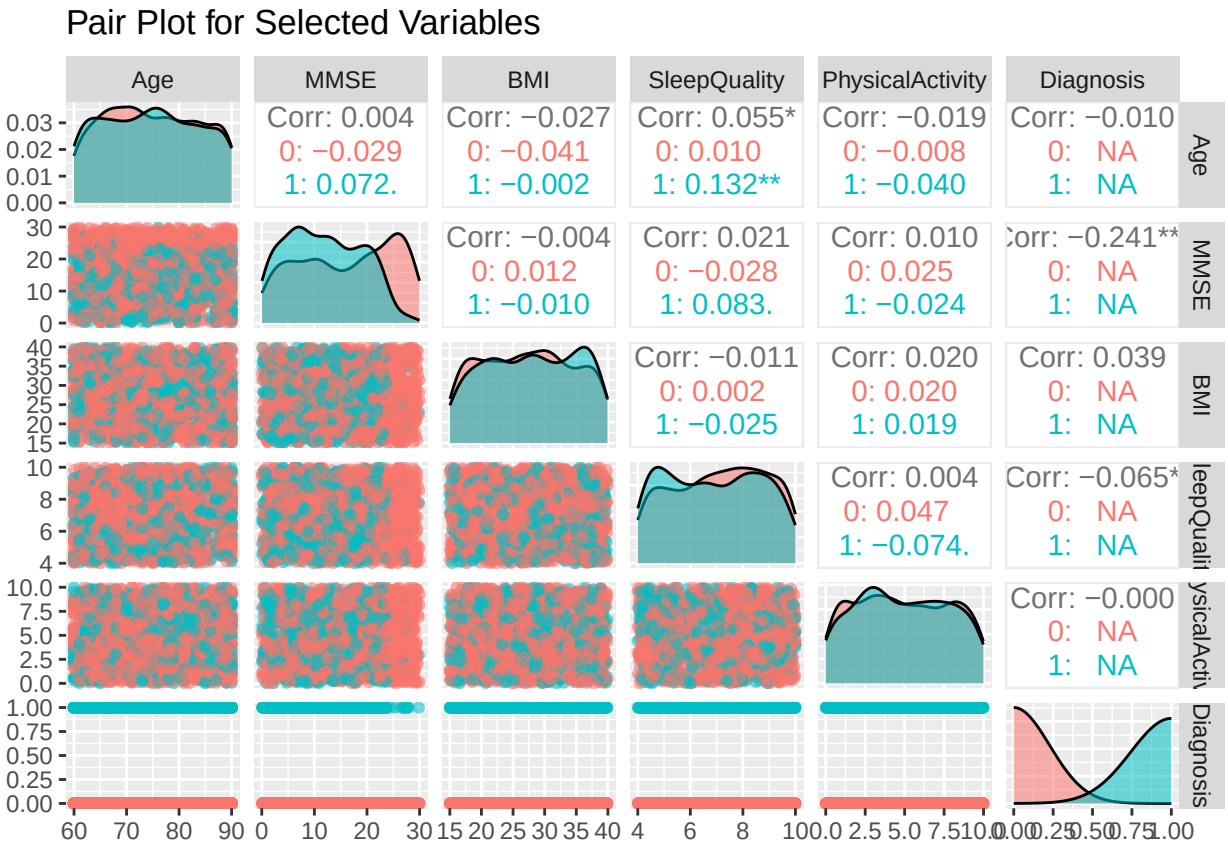

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

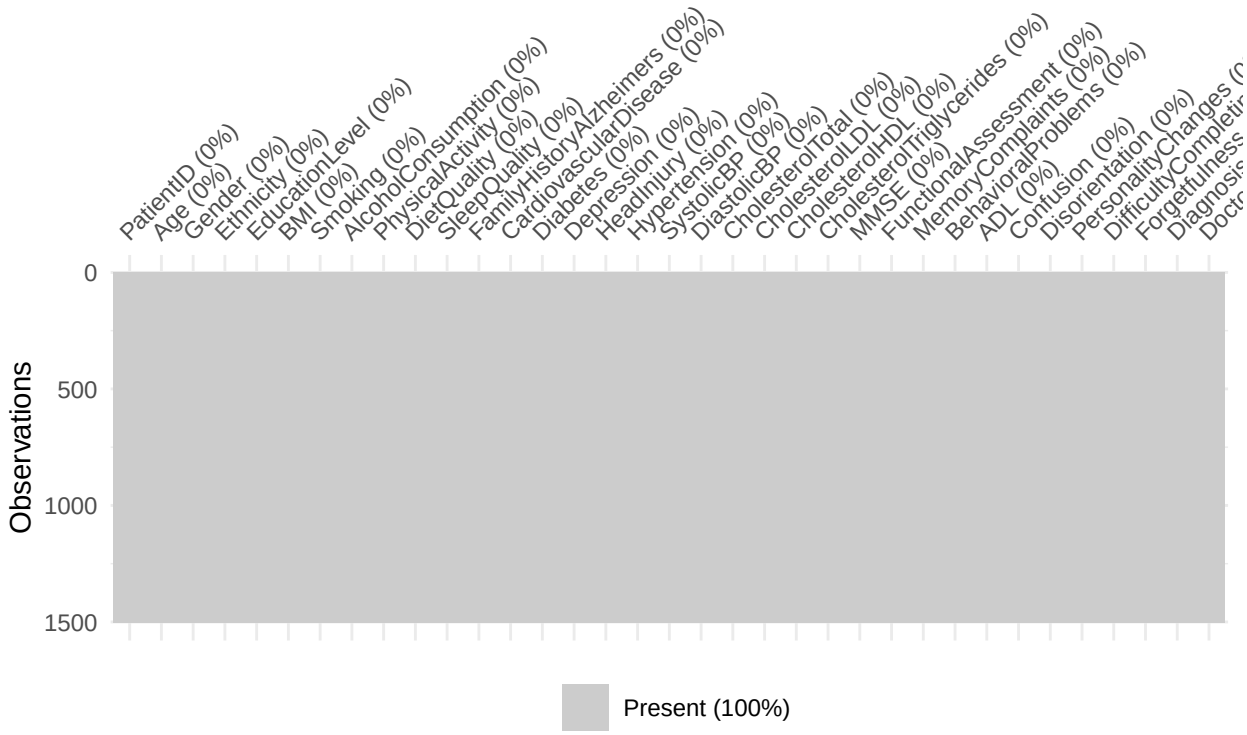
```
selected_variables <- train_data %>%
  select(Age, MMSE, BMI, SleepQuality, PhysicalActivity, Diagnosis)

ggpairs(selected_variables, aes(color = as.factor(Diagnosis), alpha = 0.5)) +
  labs(title = "Pair Plot for Selected Variables")
```

```
## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero
```

```
## Warning in cor(x, y): the standard deviation is zero

## Warning in cor(x, y): the standard deviation is zero
```




```r
library(naniar)

vis_miss(train_data) +
  labs(title = "Missing Data Visualization")
```



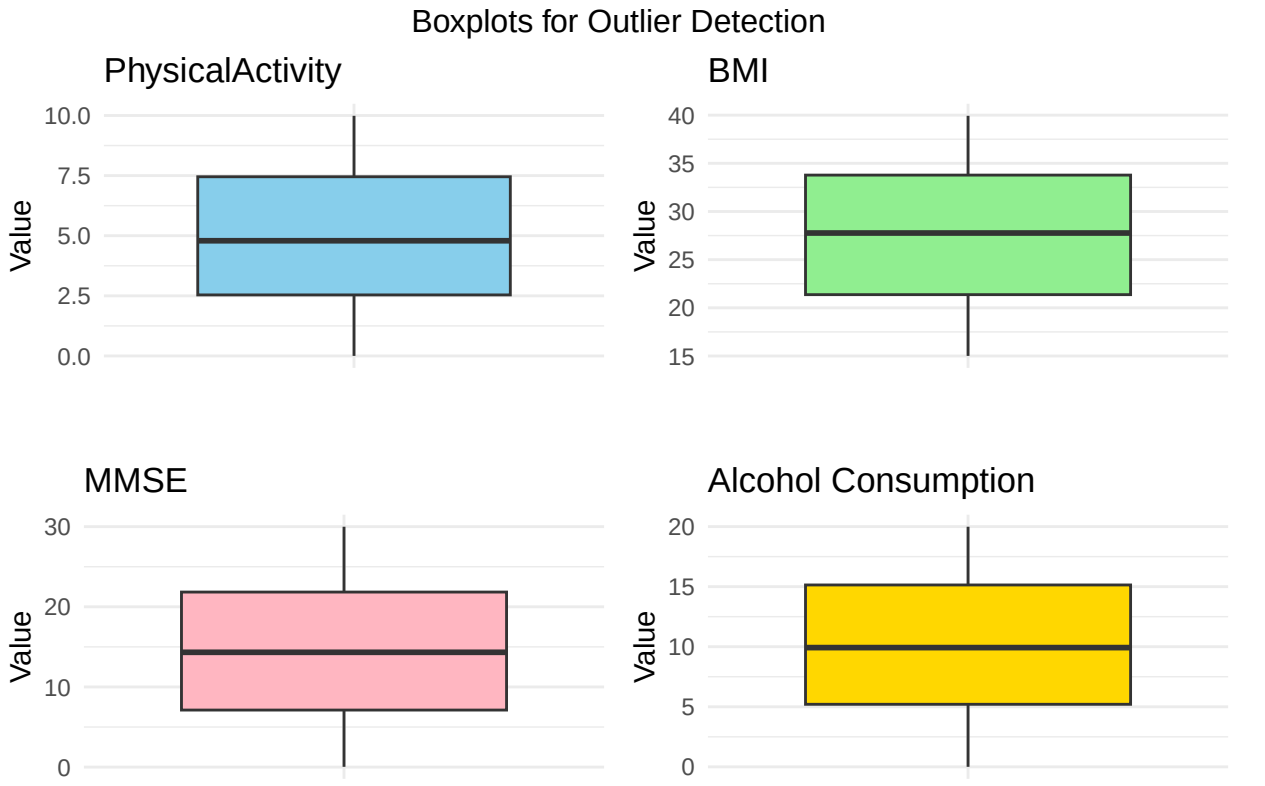

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
p1 <- ggplot(train_data, aes(x = "", y = PhysicalActivity)) +
  geom_boxplot(fill = "skyblue", outlier.color = "red") +
  labs(title = "PhysicalActivity", y = "Value", x = "") +
  theme_minimal()

p2 <- ggplot(train_data, aes(x = "", y = BMI)) +
  geom_boxplot(fill = "lightgreen", outlier.color = "red") +
  labs(title = "BMI", y = "Value", x = "") +
  theme_minimal()

p3 <- ggplot(train_data, aes(x = "", y = MMSE)) +
  geom_boxplot(fill = "lightpink", outlier.color = "red") +
  labs(title = "MMSE", y = "Value", x = "") +
  theme_minimal()

p4 <- ggplot(train_data, aes(x = "", y = AlcoholConsumption)) +
  geom_boxplot(fill = "gold", outlier.color = "red") +
  labs(title = "Alcohol Consumption", y = "Value", x = "") +
  theme_minimal()
```

```r
# Combine all boxplots into a single chart
grid.arrange(p1, p2, p3, p4, nrow = 2, top = "Boxplots for Outlier Detection")
```




```r
# Remove irrelevant columns
train_data <- train_data %>% select(-PatientID, -DoctorInCharge)  # Remove columns unlikely to provide

# Confirm Diagnosis is binary and convert to a factor if needed
if (length(unique(train_data$Diagnosis)) == 2) {
  train_data$Diagnosis <- as.factor(train_data$Diagnosis)
} else {
  stop("Diagnosis must be a binary variable.")
}

# Ensure categorical variables are factors
train_data$Gender <- as.factor(train_data$Gender)
train_data$Ethnicity <- as.factor(train_data$Ethnicity)
train_data$EducationLevel <- as.factor(train_data$EducationLevel)
```

```r
# Combine point-biserial correlation threshold and p-value testing
numeric_features <- train_data %>% select_if(is.numeric)
numeric_results <- lapply(numeric_features, function(x) {
  test <- cor.test(x, as.numeric(train_data$Diagnosis) - 1)  # Convert Diagnosis to numeric (0/1)
  list(correlation = test$estimate, p_value = test$p.value)
})

# Convert results to a data frame for filtering
numeric_df <- data.frame(
  Feature = names(numeric_results),
  Correlation = sapply(numeric_results, function(res) res$correlation),
```

```
  P_Value = sapply(numeric_results, function(res) res$p_value)
)

# Filter features based on correlation threshold and p-value
correlation_threshold <- 0.03
numeric_significant <- numeric_df %>%
  filter(abs(Correlation) > correlation_threshold & P_Value < 0.05) %>%
  pull(Feature)

# Print significant numeric features
print("Significant Numeric Features:")
```

```
## [1] "Significant Numeric Features:"
```

```
print(numeric_significant)
```

```
## [1] "SleepQuality"         "Diabetes"             "MMSE"
## [4] "FunctionalAssessment" "MemoryComplaints"     "BehavioralProblems"
## [7] "ADL"
```

```
# Combine selected numeric features and Diagnosis column
all_selected_features <- c(numeric_significant, "Diagnosis")

# Subset the dataset with the selected features
selected_data <- train_data %>%
  dplyr::select(all_of(all_selected_features))

# Display the first few rows of the selected dataset
head(selected_data)
```

```
##   SleepQuality Diabetes       MMSE FunctionalAssessment MemoryComplaints
## 1     6.744820        0  0.6946002             9.986441                1
## 2     7.568751        0 23.7899987             6.197277                0
## 3     8.247084        1  6.5920715             9.572719                0
## 4     7.666498        1 25.3426163             2.487042                0
## 5     6.231143        0  6.6277415             7.521358                1
## 6     4.764378        0 22.1699224             6.592334                0
##   BehavioralProblems      ADL Diagnosis
## 1                  0 6.009376         0
## 2                  0 7.519209         0
## 3                  0 8.573933         0
## 4                  0 6.217530         0
## 5                  0 5.193683         0
## 6                  0 9.420887         0
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
library(xgboost)
```

```
##
## Attaching package: 'xgboost'
```

```
## The following object is masked from 'package:dplyr':
##
##     slice
```

```r
set.seed(787)
trainIndex <- createDataPartition(selected_data$Diagnosis, p = 0.8, list = FALSE)
training_set <- selected_data[trainIndex, ]
validation_set <- selected_data[-trainIndex, ]
# Ensure Diagnosis is numeric
training_set$Diagnosis <- as.numeric(as.character(training_set$Diagnosis))
validation_set$Diagnosis <- as.numeric(as.character(validation_set$Diagnosis))
# Create training matrix for XGBoost
train_matrix <- xgb.DMatrix(data = as.matrix(training_set %>% dplyr::
                                               select(-Diagnosis)),
                            label = training_set$Diagnosis)
```

```r
param_grid <- expand.grid(eta = c(0.01, 0.05), max_depth = c(4, 6),
                          subsample = c(0.8, 1.0),
                          colsample_bytree = c(0.8, 1.0))
best_params <- NULL
lowest_logloss <- Inf
```

```r
for (i in 1:nrow(param_grid)) {
  params <- list(
    objective = "binary:logistic",
    eval_metric = "logloss",
    eta = param_grid$eta[i],
    max_depth = param_grid$max_depth[i],
    subsample = param_grid$subsample[i],
    colsample_bytree = param_grid$colsample_bytree[i]
  )

  set.seed(787)
  cv_results <- xgb.cv(
    params = params,
    data = train_matrix,
    nrounds = 100,
    nfold = 10,
    early_stopping_rounds = 10,
    verbose = 0
  )

  min_logloss <- min(cv_results$evaluation_log$test_logloss_mean)

  if (min_logloss < lowest_logloss) {lowest_logloss <- min_logloss
    best_params <- params
    best_nrounds <- cv_results$best_iteration
  }
}
print(best_params)
```

```
## $objective
## [1] "binary:logistic"
##
## $eval_metric
## [1] "logloss"
##
## $eta
```

```
## [1] 0.05
##
## $max_depth
## [1] 6
##
## $subsample
## [1] 0.8
##
## $colsample_bytree
## [1] 1
```

```
print(paste("Best Number of Rounds:", best_nrounds))
```

```
## [1] "Best Number of Rounds: 85"
```

```
final_model <- xgb.train(params = best_params, data = train_matrix,
                         nrounds = best_nrounds)
```

```
valid_matrix <- xgb.DMatrix(data = as.matrix(validation_set %>% dplyr::
                                               select(-Diagnosis)))
final_preds <- predict(final_model, valid_matrix)
final_class <- ifelse(final_preds > 0.5, 1, 0)
confusionMatrix(factor(final_class), factor(validation_set$Diagnosis))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 187   8
##          1   7  98
##
##                Accuracy : 0.95
##                  95% CI : (0.9189, 0.9717)
##     No Information Rate : 0.6467
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.8904
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.9639
##             Specificity : 0.9245
##          Pos Pred Value : 0.9590
##          Neg Pred Value : 0.9333
##              Prevalence : 0.6467
##          Detection Rate : 0.6233
##    Detection Prevalence : 0.6500
##       Balanced Accuracy : 0.9442
##
##        'Positive' Class : 0
##
```

```
valid_preds_prob <- predict(final_model, valid_matrix)
thresholds <- seq(0.05, 0.95, by = 0.05)
best_threshold <- 0
best_f1 <- 0
```

```r
for (thresh in thresholds) {
valid_preds_class <- ifelse(valid_preds_prob > thresh, 1, 0)
cm <- confusionMatrix(factor(valid_preds_class),factor(validation_set$Diagnosis))
precision <- cm$byClass["Pos Pred Value"]
recall <- cm$byClass["Sensitivity"]
f1 <- 2 * (precision * recall) / (precision + recall)
if (!is.na(f1) && f1 > best_f1) {
best_f1 <- f1
best_threshold <- thresh
}
}
print(paste("Best Threshold:", best_threshold))
```

```
## [1] "Best Threshold: 0.35"
```

```r
print(paste("Best F1-Score:", best_f1))
```

```
## [1] "Best F1-Score: 0.963917525773196"
```

```r
selected_features_test <- all_selected_features[all_selected_features !=
                                                  "Diagnosis"]
test_matrix <- xgb.DMatrix(data = as.matrix(test_data %>% dplyr::
                                              select(all_of(selected_features_test))))
test_preds <- predict(final_model, test_matrix)
test_class <- ifelse(test_preds > best_threshold, 1, 0)
submission <- data.frame(PatientID = test_data$PatientID, Diagnosis = test_class)
write.csv(submission, "prediction.csv", row.names = FALSE)
```