#HEALTH INSURANCE FRAUD DETECTION

Mohammed Ba-Aoum

```r
# Load libraries
library(readxl)
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)
library(caret)

## Loading required package: lattice

library(rpart)
library(rpart.plot)
library(randomForest)

## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin

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

# Load the dataset
file_path <- "Health_Insurance_Fraud.xlsx"
data <- read_excel(file_path, sheet = "Fraud_Detection_decsion tree")

# Explore the data
# Clean column names (e.g., replace invalid ones)
```

```r
names(data) <- make.names(names(data))
str(data)
```

```
## tibble [1,000 × 39] (S3: tbl_df/tbl/data.frame)
##  $ months_as_customer         : num [1:1000] 328 228 134 256 228 256 137
165 27 212 ...
##  $ age                        : num [1:1000] 48 42 29 41 44 39 34 37 33 42
...
##  $ policy_number              : num [1:1000] 521585 342868 687698 227811
367455 ...
##  $ policy_bind_date           : POSIXct[1:1000], format: "2014-10-17"
"2006-06-27" ...
##  $ policy_state               : chr [1:1000] "OH" "IN" "OH" "IL" ...
##  $ policy_csl                 : chr [1:1000] "250/500" "250/500" "100/300"
"250/500" ...
##  $ policy_deductable          : num [1:1000] 1000 2000 2000 2000 1000 1000
1000 1000 500 500 ...
##  $ policy_annual_premium      : num [1:1000] 1407 1197 1413 1416 1584 ...
##  $ umbrella_limit             : num [1:1000] 0e+00 5e+06 5e+06 6e+06 6e+06
0e+00 0e+00 0e+00 0e+00 0e+00 ...
##  $ insured_zip                : num [1:1000] 466132 468176 430632 608117
610706 ...
##  $ insured_sex                : chr [1:1000] "MALE" "MALE" "FEMALE"
"FEMALE" ...
##  $ insured_education_level    : chr [1:1000] "MD" "MD" "PhD" "PhD" ...
##  $ insured_occupation         : chr [1:1000] "craft-repair" "machine-op-
inspct" "sales" "armed-forces" ...
##  $ insured_hobbies            : chr [1:1000] "sleeping" "reading" "board-
games" "board-games" ...
##  $ insured_relationship       : chr [1:1000] "husband" "other-relative"
"own-child" "unmarried" ...
##  $ capital.gains              : num [1:1000] 53300 0 35100 48900 66000 0 0
0 0 0 ...
##  $ capital.loss               : num [1:1000] 0 0 0 -62400 -46000 0 -77000
0 0 -39300 ...
##  $ incident_date              : POSIXct[1:1000], format: "2015-01-25"
"2015-01-21" ...
##  $ incident_type              : chr [1:1000] "Single Vehicle Collision"
"Vehicle Theft" "Multi-vehicle Collision" "Single Vehicle Collision" ...
##  $ collision_type             : chr [1:1000] "Side Collision" "?" "Rear
Collision" "Front Collision" ...
##  $ incident_severity          : chr [1:1000] "Major Damage" "Minor Damage"
"Minor Damage" "Major Damage" ...
##  $ authorities_contacted      : chr [1:1000] "Police" "Police" "Police"
"Police" ...
##  $ incident_state             : chr [1:1000] "SC" "VA" "NY" "OH" ...
##  $ incident_city              : chr [1:1000] "Columbus" "Riverwood"
"Columbus" "Arlington" ...
##  $ incident_location          : chr [1:1000] "9935 4th Drive" "6608 MLK
Hwy" "7121 Francis Lane" "6956 Maple Drive" ...
```

```
##  $ incident_hour_of_the_day   : num [1:1000] 5 8 7 5 20 19 0 23 21 14 ...
##  $ number_of_vehicles_involved: num [1:1000] 1 1 3 1 1 3 3 3 1 1 ...
##  $ property_damage            : chr [1:1000] "YES" "?" "NO" "?" ...
##  $ bodily_injuries            : num [1:1000] 1 0 2 1 0 0 0 2 1 2 ...
##  $ witnesses                  : num [1:1000] 2 0 3 2 1 2 0 2 1 1 ...
##  $ police_report_available    : chr [1:1000] "YES" "?" "NO" "NO" ...
##  $ total_claim_amount         : num [1:1000] 71610 5070 34650 63400 6500
...
##  $ injury_claim               : num [1:1000] 6510 780 7700 6340 1300 ...
##  $ property_claim             : num [1:1000] 13020 780 3850 6340 650 ...
##  $ vehicle_claim              : num [1:1000] 52080 3510 23100 50720 4550
...
##  $ auto_make                  : chr [1:1000] "Saab" "Mercedes" "Dodge"
"Chevrolet" ...
##  $ auto_model                 : chr [1:1000] "92x" "E400" "RAM" "Tahoe"
...
##  $ auto_year                  : num [1:1000] 2004 2007 2007 2014 2009 ...
##  $ fraud_reported             : chr [1:1000] "Y" "Y" "N" "Y" ...
```

```
summary(data)
```

```
##  months_as_customer      age         policy_number
##  Min.   :  0.0      Min.   :19.00   Min.   :100804
##  1st Qu.:115.8      1st Qu.:32.00   1st Qu.:335980
##  Median :199.5      Median :38.00   Median :533135
##  Mean   :204.0      Mean   :38.95   Mean   :546239
##  3rd Qu.:276.2      3rd Qu.:44.00   3rd Qu.:759100
##  Max.   :479.0      Max.   :64.00   Max.   :999435
##  policy_bind_date              policy_state        policy_csl
##  Min.   :1990-01-08 00:00:00   Length:1000        Length:1000
##  1st Qu.:1995-09-19 00:00:00   Class :character   Class :character
##  Median :2002-04-01 12:00:00   Mode  :character   Mode  :character
##  Mean   :2002-02-08 04:40:48
##  3rd Qu.:2008-04-21 12:00:00
##  Max.   :2015-02-22 00:00:00
##  policy_deductable policy_annual_premium umbrella_limit      insured_zip
##  Min.   : 500      Min.   : 433.3        Min.   :-1000000   Min.   :430104
##  1st Qu.: 500      1st Qu.:1089.6        1st Qu.:       0   1st Qu.:448404
##  Median :1000      Median :1257.2        Median :       0   Median :466446
##  Mean   :1136      Mean   :1256.4        Mean   : 1101000   Mean   :501214
##  3rd Qu.:2000      3rd Qu.:1415.7        3rd Qu.:       0   3rd Qu.:603251
##  Max.   :2000      Max.   :2047.6        Max.   :10000000   Max.   :620962
##  insured_sex        insured_education_level insured_occupation
##  Length:1000        Length:1000             Length:1000
##  Class :character   Class :character        Class :character
##  Mode  :character   Mode  :character        Mode  :character
##
##
##
##  insured_hobbies    insured_relationship capital.gains     capital.loss
```

```
##  Length:1000        Length:1000          Min.   :     0   Min.   :-111100
##  Class :character   Class :character     1st Qu.:     0   1st Qu.: -51500
##  Mode  :character   Mode  :character     Median :     0   Median : -23250
##                                         Mean   : 25126   Mean   : -26794
##                                         3rd Qu.: 51025   3rd Qu.:      0
##                                         Max.   :100500   Max.   :      0
##  incident_date                incident_type      collision_type
##  Min.   :2015-01-01 00:00:00   Length:1000        Length:1000
##  1st Qu.:2015-01-15 00:00:00   Class :character   Class :character
##  Median :2015-01-31 00:00:00   Mode  :character   Mode  :character
##  Mean   :2015-01-30 08:02:24
##  3rd Qu.:2015-02-15 00:00:00
##  Max.   :2015-03-01 00:00:00
##  incident_severity  authorities_contacted incident_state     incident_city
##  Length:1000        Length:1000           Length:1000        Length:1000
##  Class :character   Class :character      Class :character   Class
:character
##  Mode  :character   Mode  :character      Mode  :character   Mode
:character
##
##
##
##  incident_location  incident_hour_of_the_day number_of_vehicles_involved
##  Length:1000        Min.   : 0.00            Min.   :1.000
##  Class :character   1st Qu.: 6.00            1st Qu.:1.000
##  Mode  :character   Median :12.00            Median :1.000
##                     Mean   :11.64            Mean   :1.839
##                     3rd Qu.:17.00            3rd Qu.:3.000
##                     Max.   :23.00            Max.   :4.000
##  property_damage    bodily_injuries   witnesses
police_report_available
##  Length:1000        Min.   :0.000   Min.   :0.000   Length:1000
##  Class :character   1st Qu.:0.000   1st Qu.:1.000   Class :character
##  Mode  :character   Median :1.000   Median :1.000   Mode  :character
##                     Mean   :0.992   Mean   :1.487
##                     3rd Qu.:2.000   3rd Qu.:2.000
##                     Max.   :2.000   Max.   :3.000
##  total_claim_amount  injury_claim   property_claim  vehicle_claim
##  Min.   :   100     Min.   :    0   Min.   :    0   Min.   :   70
##  1st Qu.: 41812     1st Qu.: 4295   1st Qu.: 4445   1st Qu.:30292
##  Median : 58055     Median : 6775   Median : 6750   Median :42100
##  Mean   : 52762     Mean   : 7433   Mean   : 7400   Mean   :37929
##  3rd Qu.: 70592     3rd Qu.:11305   3rd Qu.:10885   3rd Qu.:50822
##  Max.   :114920     Max.   :21450   Max.   :23670   Max.   :79560
##   auto_make          auto_model          auto_year    fraud_reported
##  Length:1000        Length:1000        Min.   :1995   Length:1000
##  Class :character   Class :character   1st Qu.:2000   Class :character
##  Mode  :character   Mode  :character   Median :2005   Mode  :character
##                                        Mean   :2005
```

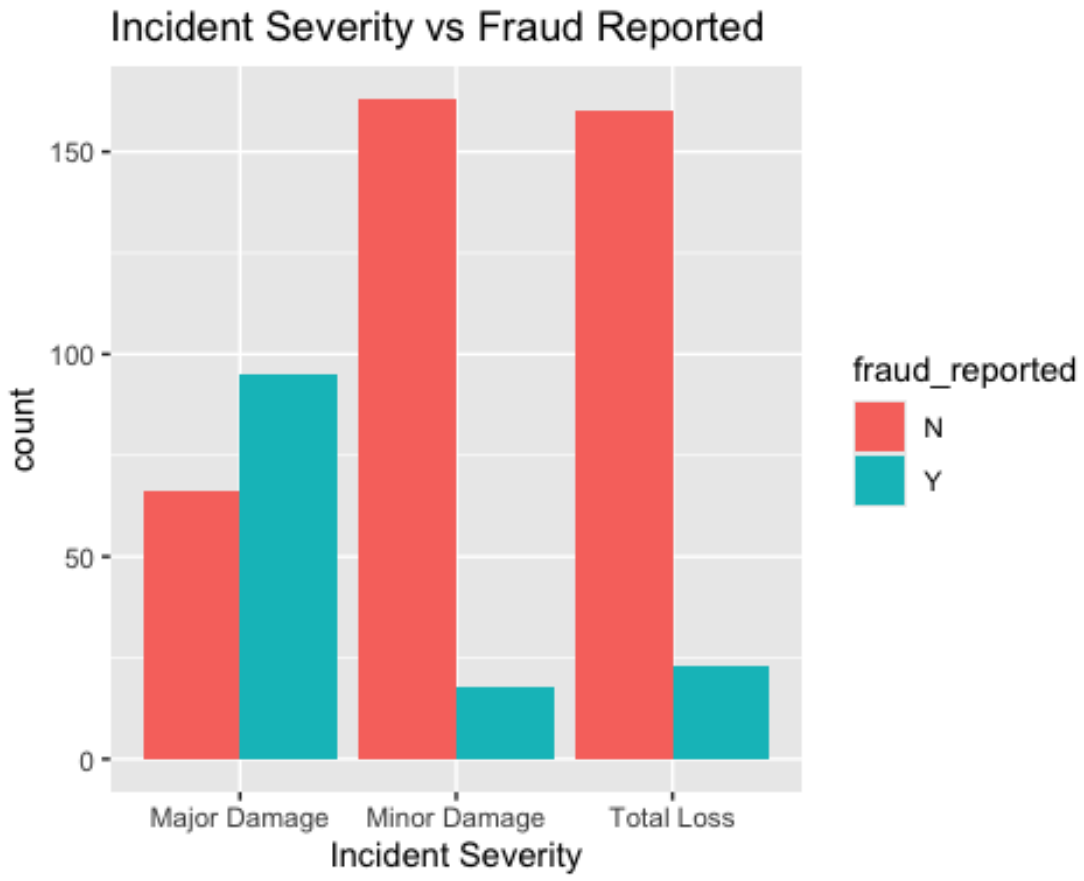
```
##                                          3rd Qu.:2010
##                                          Max.   :2015

# Convert relevant columns to factors
factor_vars <- c("fraud_reported", "police_report_available",
                 "policy_state", "auto_make", "auto_model", "policy_csl")
data[factor_vars] <- lapply(data[factor_vars], as.factor)

# Replace "?" with NA only in character columns
char_cols <- sapply(data, is.character)
data[char_cols] <- lapply(data[char_cols], function(x) ifelse(x == "?", NA,
x))
data <- na.omit(data)  # Drop rows with missing values

# Drop unnecessary columns

data_clean <- data %>%
  select(-c(policy_number, policy_bind_date, insured_zip, incident_location))
```

EDA Visualizations

```
# 1. Fraud report distribution
ggplot(data_clean, aes(x = fraud_reported, fill = fraud_reported)) +
  geom_bar() +
  labs(title = "Fraud Reported Distribution", x = "Fraud Reported", y =
"Count")
```

#Insight: This plot shows that a majority of the insurance claims in the dataset are non-fraudulent (N), but a significant proportion (around 25%) are reported as fraudulent (Y). This class imbalance needs to be kept in mind when modeling, as it may impact model sensitivity and specificity.

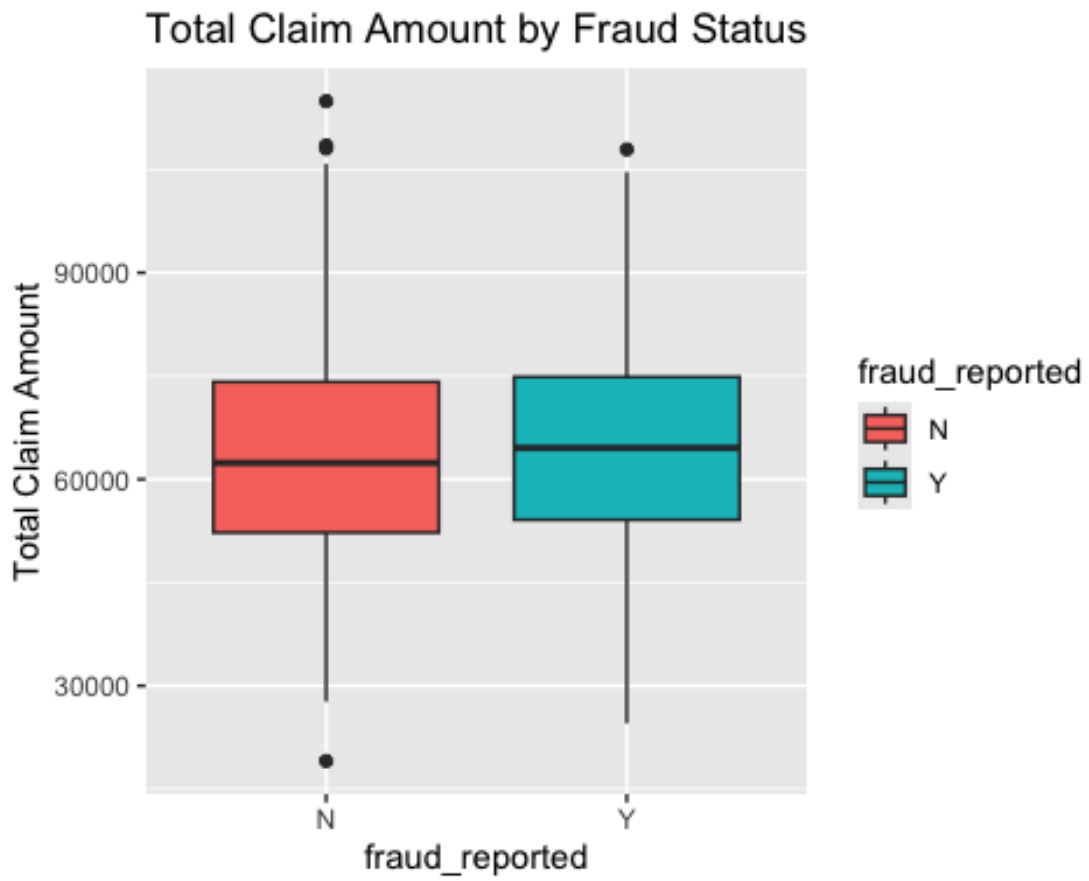
```
# 2. Incident severity by fraud status
ggplot(data_clean, aes(x = incident_severity, fill = fraud_reported)) +
  geom_bar(position = "dodge") +
  labs(title = "Incident Severity vs Fraud Reported", x = "Incident
Severity")
```



#

Insight: Claims involving “Major Damage” are more frequently reported as fraudulent compared to those with “Minor” or “Trivial” damage. This suggests that claim severity is a strong indicator of potential fraud, likely because larger claims present a higher opportunity for abuse.

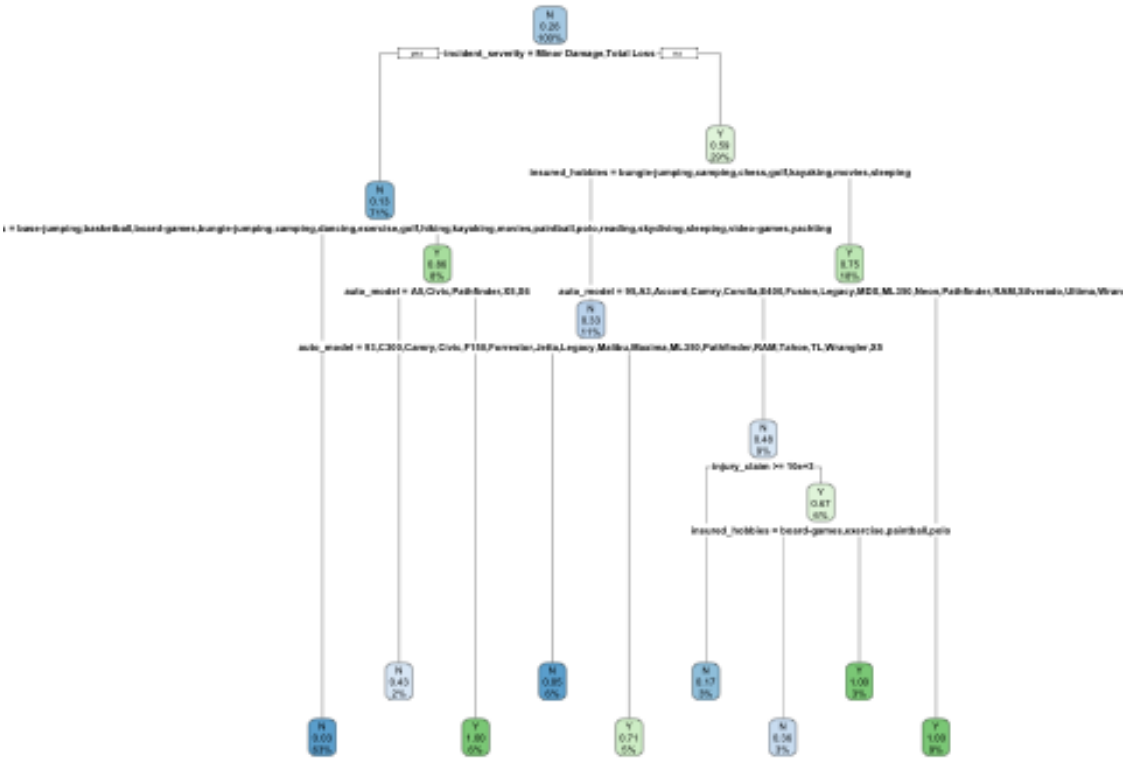
```
# 3. Boxplot of total claim amount by fraud
ggplot(data_clean, aes(x = fraud_reported, y = total_claim_amount, fill =
fraud_reported)) +
  geom_boxplot() +
  labs(title = "Total Claim Amount by Fraud Status", y = "Total Claim
Amount")
```



#

Insight: The boxplot reveals that fraudulent claims tend to have a wider range and higher median total claim amounts. This supports the intuition that fraudsters often inflate claim values to maximize gain, making this feature particularly valuable for prediction.

```
# Split data into training and testing sets
set.seed(123)
trainIndex <- createDataPartition(data_clean$fraud_reported, p = 0.7, list =
FALSE)
train <- data_clean[trainIndex, ]
test <- data_clean[-trainIndex, ]

# Build a Decision Tree model
tree_model <- rpart(fraud_reported ~ ., data = train, method = "class")
rpart.plot(tree_model)
```



```r
# Predict on test data
predictions <- predict(tree_model, test, type = "class")

# Evaluate model performance
confusionMatrix(predictions, test$fraud_reported)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   N   Y
##          N 103  20
##          Y  13  20
##
##                Accuracy : 0.7885
##                  95% CI : (0.7159, 0.8497)
##     No Information Rate : 0.7436
##     P-Value [Acc > NIR] : 0.1152
##
##                   Kappa : 0.4115
##
##  Mcnemar's Test P-Value : 0.2963
##
##             Sensitivity : 0.8879
```

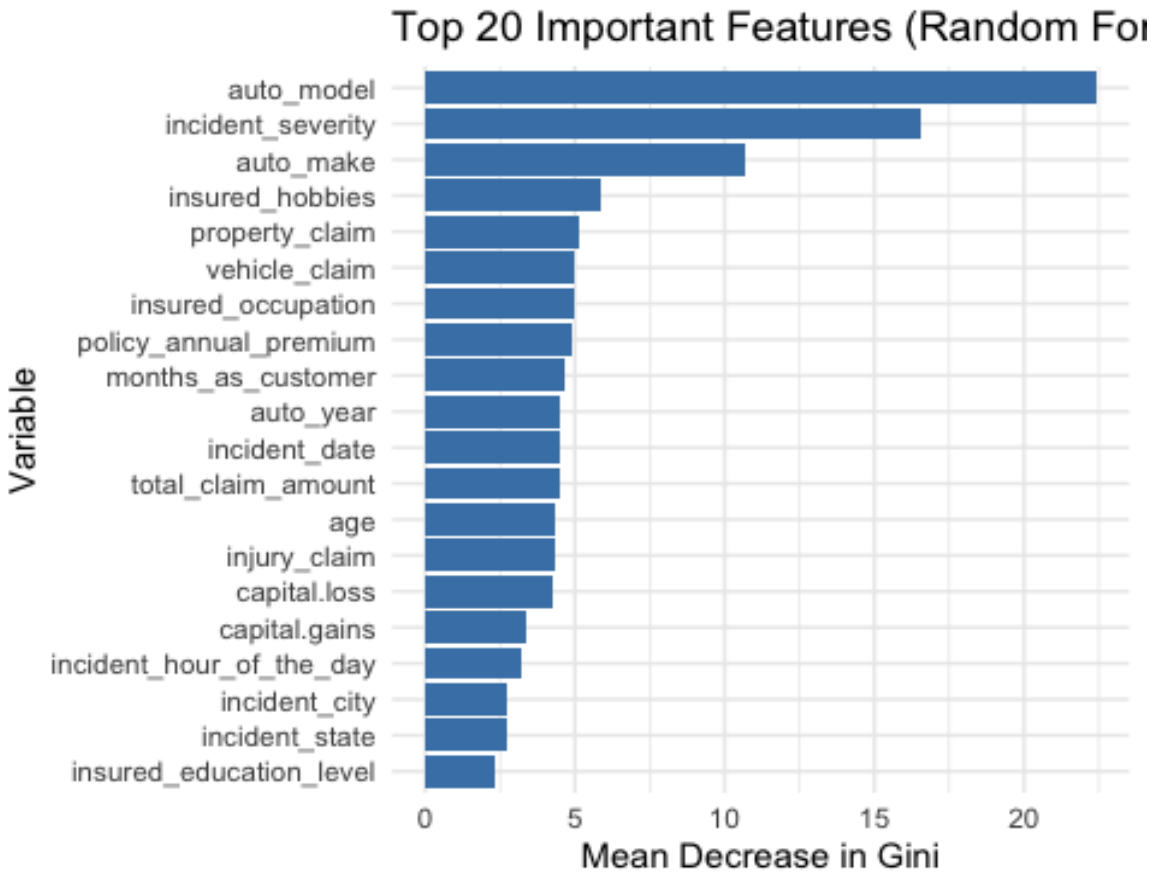
```
##             Specificity : 0.5000
##          Pos Pred Value : 0.8374
##          Neg Pred Value : 0.6061
##              Prevalence : 0.7436
##          Detection Rate : 0.6603
##    Detection Prevalence : 0.7885
##       Balanced Accuracy : 0.6940
##
##        'Positive' Class : N
##
```

```
# Random Forst

rf_model <- randomForest(fraud_reported ~ ., data = train, ntree = 100,
importance = TRUE)
rf_pred <- predict(rf_model, test)
confusionMatrix(rf_pred, test$fraud_reported)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   N   Y
##          N 106  26
##          Y  10  14
##
##                Accuracy : 0.7692
##                  95% CI : (0.6951, 0.8328)
##     No Information Rate : 0.7436
##     P-Value [Acc > NIR] : 0.26345
##
##                   Kappa : 0.3036
##
##  Mcnemar's Test P-Value : 0.01242
##
##             Sensitivity : 0.9138
##             Specificity : 0.3500
##          Pos Pred Value : 0.8030
##          Neg Pred Value : 0.5833
##              Prevalence : 0.7436
##          Detection Rate : 0.6795
##    Detection Prevalence : 0.8462
##       Balanced Accuracy : 0.6319
##
##        'Positive' Class : N
##
```

```
#Feature Importance
# Extract variable importance
importance_df <- as.data.frame(importance(rf_model))
importance_df$Variable <- rownames(importance_df)
```

```r
importance_df %>%
  arrange(desc(MeanDecreaseGini)) %>%
  top_n(20, MeanDecreaseGini) %>%
  ggplot(aes(x = reorder(Variable, MeanDecreaseGini), y = MeanDecreaseGini))
+
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(title = "Top 20 Important Features (Random Forest)",
       x = "Variable", y = "Mean Decrease in Gini") +
  theme_minimal()
```



Insight: The random forest model identifies features like incident_severity, auto_model, auto_make, insured_hobbies, and property_claim as some of the most important for detecting fraud. These variables offer valuable behavioral and contextual cues, showing that not only financial metrics but also customer behavior patterns contribute to effective fraud detection.