Report: Bank Marketing Data Analysis and Predictive Modeling

Introduction

This report outlines the steps taken to analyze the Bank Marketing dataset and build a predictive model to determine whether a client will subscribe to a term deposit. The dataset was obtained from the UCI Machine Learning Repository and contains information about bank clients, including demographic data, financial information, and marketing campaign details.

Project Overview

The project involves the following key steps:

1. **Data Downloading and Loading**: The dataset was downloaded from the UCI Machine Learning Repository and loaded into R.
2. **Data Exploration**: Initial exploration of the dataset to understand its structure and content.
3. **Data Preprocessing**: Cleaning and preparing the data for analysis, including handling categorical variables and checking for missing values.
4. **Exploratory Data Analysis (EDA)**: Univariate and bivariate analysis to uncover patterns and relationships in the data.
5. **Feature Engineering**: Creating new features to improve model performance.
6. **Model Building**: Training a Random Forest model to predict term deposit subscriptions.
7. **Model Evaluation**: Assessing the model's performance using a confusion matrix and ROC curve.
8. **Deployment**: Saving the model for future use.

Detailed Steps

1. Data Downloading and Loading

The dataset was downloaded from the UCI Machine Learning Repository using the provided URL. The dataset was then extracted and loaded into R as a data frame.

```
# Download and load the dataset

bank_data_url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00222/bank-additional.zip"

dest_file <- "bank-additional.zip"

download.file(bank_data_url, destfile = dest_file, mode = "wb")
```

```
unzip(dest_file, exdir = "bank-data")

csv_file <- "bank-data/bank-additional/bank-additional-full.csv"

data <- read.csv(csv_file, sep = ";")
```

2. Data Exploration

The initial exploration involved displaying the first few rows of the dataset, checking its structure, and summarizing its content. The dataset contains 20 variables, including both numerical and categorical data.

```
# Display the first few rows

print(head(data))


# Check the structure of the dataset

print(str(data))


# Get a summary of the dataset

print(summary(data))


# Check for missing values

print(sum(is.na(data)))
```

3. Data Preprocessing

Categorical variables were converted to factors to facilitate analysis. The dataset was checked for missing values, and none were found.

```
# Convert categorical variables to factors

data$job <- as.factor(data$job)

data$marital <- as.factor(data$marital)

data$education <- as.factor(data$education)

data$default <- as.factor(data$default)

data$housing <- as.factor(data$housing)

data$loan <- as.factor(data$loan)

data$contact <- as.factor(data$contact)
```

```
data$month <- as.factor(data$month)

data$poutcome <- as.factor(data$poutcome)

data$y <- as.factor(data$y)


# Check for missing values

sum(is.na(data))
```

4. Exploratory Data Analysis (EDA)

Univariate and bivariate analyses were conducted to understand the distribution of variables and their relationships with the target variable (y).

```
# Univariate analysis

ggplot(data, aes(x = age)) + geom_histogram(binwidth = 5, fill = "blue")

ggplot(data, aes(x = balance)) + geom_histogram(binwidth = 1000, fill = "green")


# Bivariate analysis

ggplot(data, aes(x = job, fill = y)) + geom_bar(position = "fill")

ggplot(data, aes(x = education, fill = y)) + geom_bar(position = "fill")
```

5. Feature Engineering

A new feature, age_group, was created to categorize clients into different age groups.

```
# Create age groups

data <- data %>%

  mutate(age_group = cut(age, breaks = c(0, 30, 40, 50, 60, 100), labels = c("<30", "30-40", "40-50", "50-60", "60+"))
```

6. Model Building

The dataset was split into training and testing sets, and a Random Forest model was trained on the training data.

```
# Split the data into training and testing sets

set.seed(123)

trainIndex <- createDataPartition(data$y, p = 0.8, list = FALSE)

trainData <- data[trainIndex, ]
```

```
testData <- data[-trainIndex, ]


# Train a Random Forest model
model <- randomForest(y ~ ., data = trainData, ntree = 100, importance = TRUE)
```

7. Model Evaluation

The model's performance was evaluated using a confusion matrix and ROC curve. The AUC (Area Under the Curve) was calculated to assess the model's predictive power.

```
# Make predictions on the test set
predictions <- predict(model, testData)


# Confusion matrix
confusionMatrix(predictions, testData$y)


# ROC curve and AUC
roc_curve <- roc(testData$y, as.numeric(predictions))
plot(roc_curve)
auc(roc_curve)
```

8. Deployment

The trained model was saved for future use.

```
# Save the model
saveRDS(model, "term_deposit_model.rds")
```

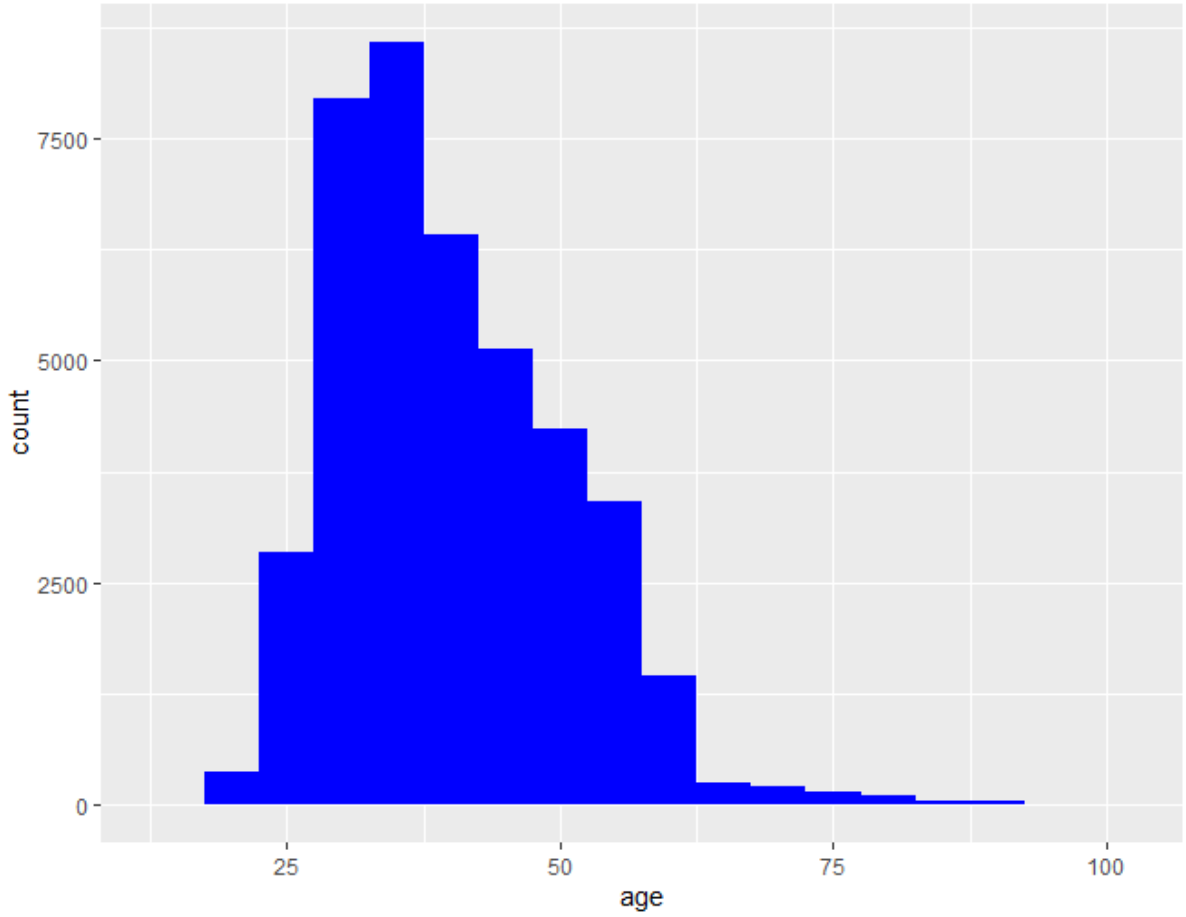
Conclusion

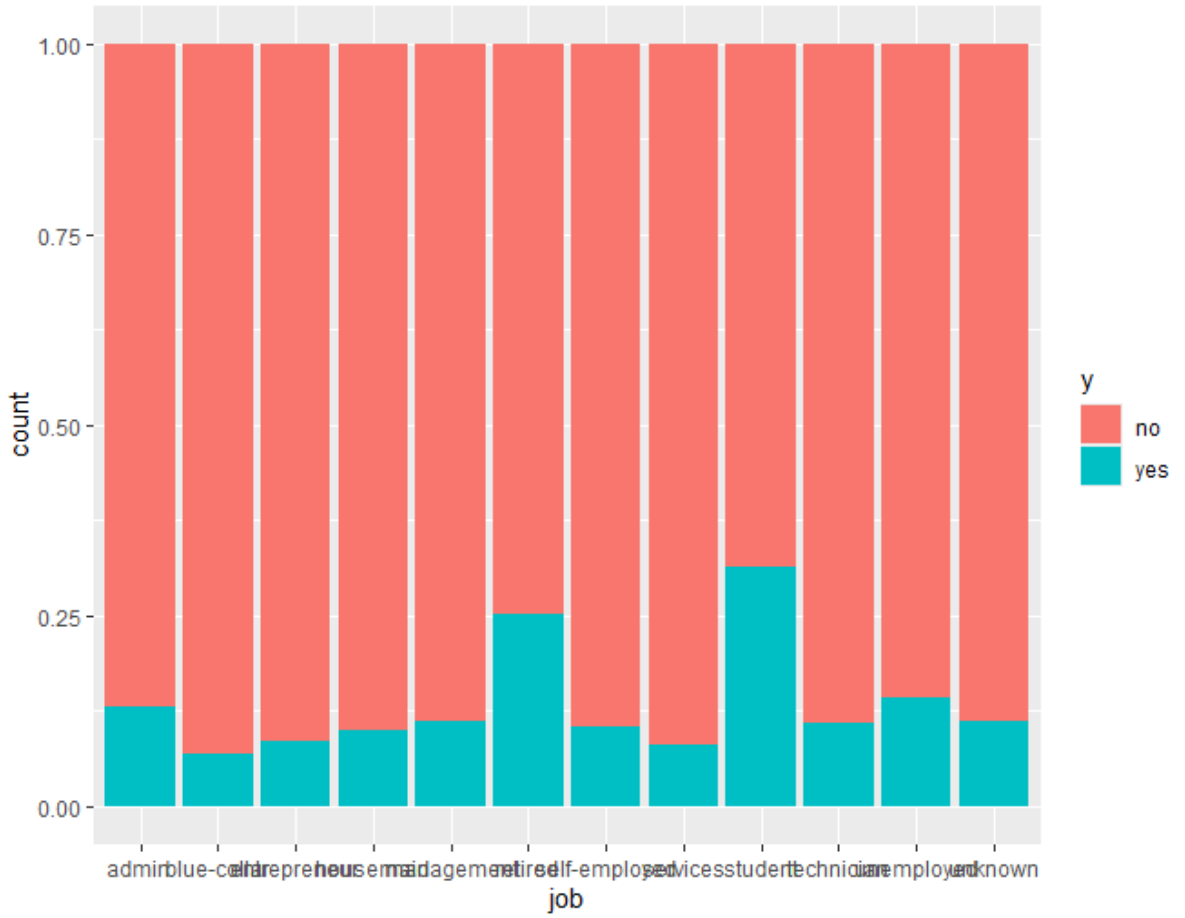
The project successfully analyzed the Bank Marketing dataset and built a predictive model using Random Forest. The model's performance was evaluated, and it demonstrated good predictive power, as indicated by the ROC curve and AUC. The model was saved for future deployment, allowing for potential use in real-world scenarios to predict term deposit subscriptions.

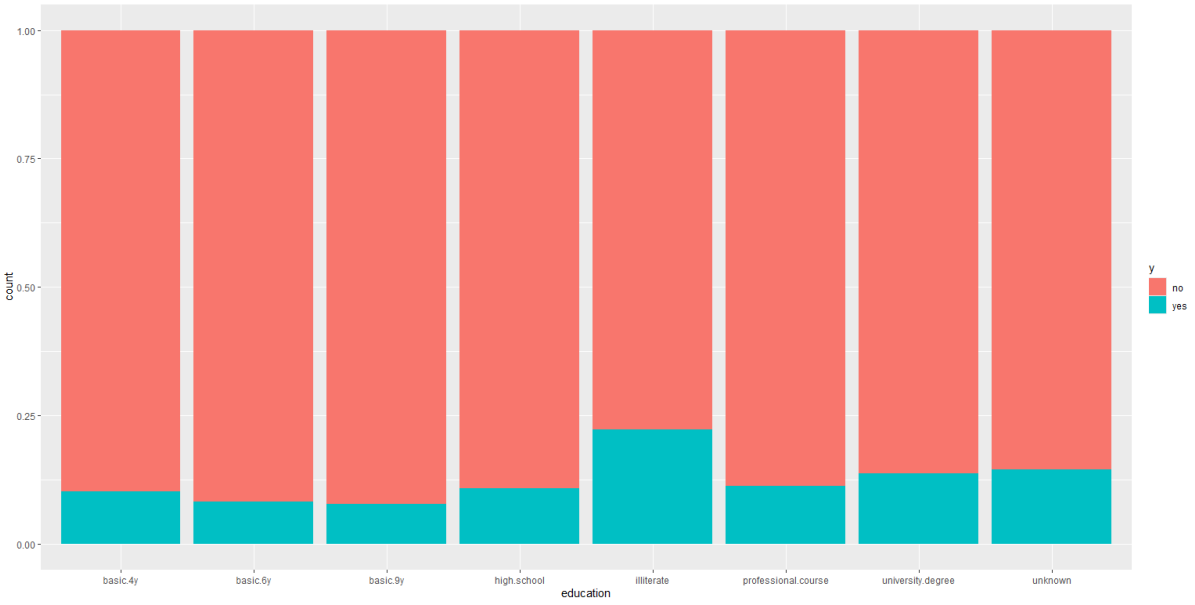
Future Work

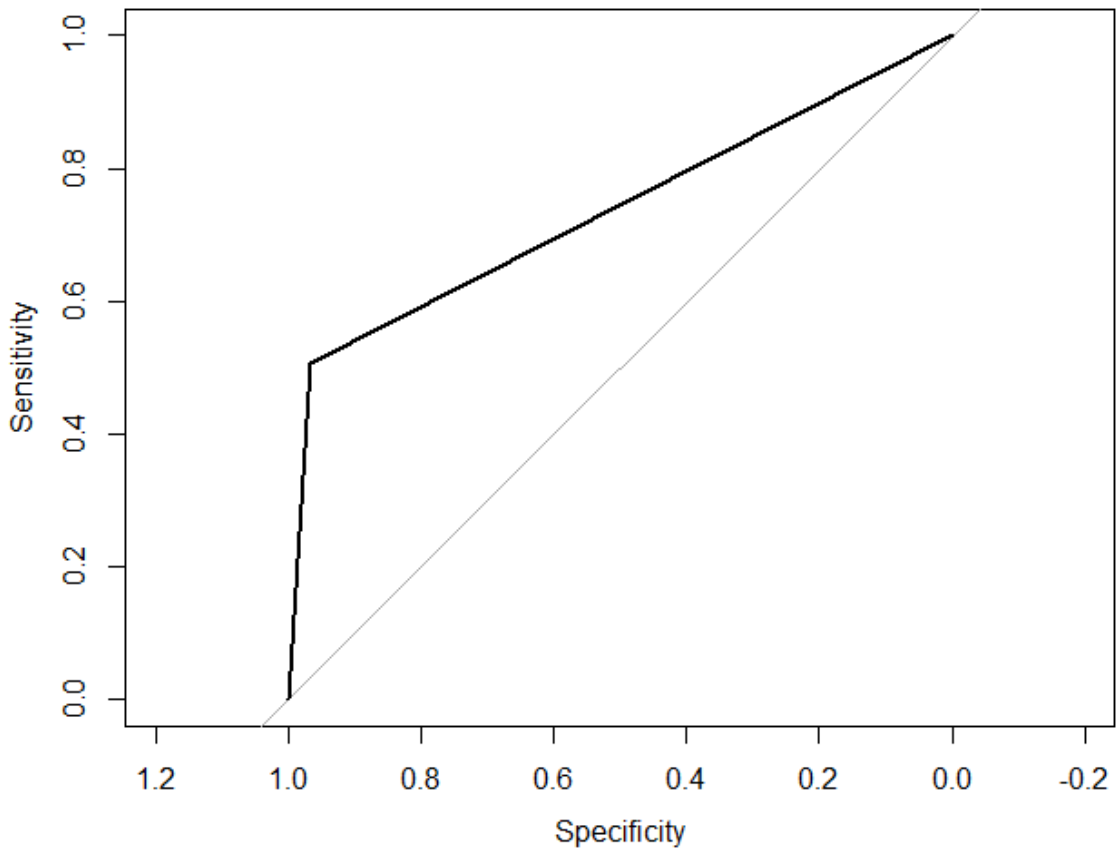
- **Hyperparameter Tuning**: Further tuning of the Random Forest model parameters to improve performance.

- **Feature Selection**: Exploring additional feature engineering techniques to enhance model accuracy.
- **Model Comparison**: Comparing the Random Forest model with other machine learning algorithms, such as logistic regression or gradient boosting.




1.00
0.75
0.50
0.25
0.00
count
job
y
no
yes


1.00
0.75
0.50
0.25
0.00
count
basic.4y
basic.6y
basic.9y
high.school
illiterate
professional.course
university.degree
unknown
education
y
no
yes


Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
Specificity