**Algorithm and Dataset**

Description of Project:

This project aims to predict customer churn in the telecommunications industry using decision tree algorithms. The objective is to identify customers at high risk of churning based on historical data, which helps in developing effective retention strategies.

Algorithm and Code:

1. **Load Libraries:**

* **Input:** None
* **Action:** Install and load necessary libraries.
* **Libraries Used:**

install.packages(c("caret", "repr", "caTools", "rpart", "rpart.plot", "ggpubr", "ggcorrplot"))

library(tidyverse)

library(caret)

library(repr)

library(caTools)

library(rpart)

library(rpart.plot)

library(ggpubr)

library(ggcorrplot)

1. **Load and Prepare Data:**

 **Input:** CSV file with customer data.

 **Action:**

* Load dataset.
* Clean data by removing rows with missing values.
* Preprocess categorical features (e.g., replace specific service labels).
* Convert columns to appropriate data types.

**Code:**

# Load data

data <- read.csv("Telco-Customer-Churn.csv")

# Remove rows with missing values

data <- data[complete.cases(data), ]

# Preprocess categorical features

data <- data.frame(lapply(data, function(x) gsub("No internet service", "No", x)))

data <- data.frame(lapply(data, function(x) gsub("No phone service", "No", x)))

# Convert columns to appropriate data types

data$SeniorCitizen <- as.factor(ifelse(data$SeniorCitizen == 1, 'YES', 'NO'))

num\_columns <- c("tenure", "MonthlyCharges", "TotalCharges")

data[num\_columns] <- sapply(data[num\_columns], as.numeric)

1. **Feature Engineering:**

 **Input:** Cleaned and processed data.

 **Action:**

* Create new features (e.g., categorize tenure).
* Convert categorical variables to dummy variables.

 **Code:**

# Feature Engineering

data <- mutate(data, tenure\_year = tenure)

data$tenure\_year <- cut(data$tenure\_year,

breaks = c(-Inf, 12, 24, 36, 48, 60, 72, Inf),

labels = c('0-1 year', '1-2 years', '2-3 years', '3-4 years', '4-5 years', '5-6 years'))

# Convert categorical variables to dummy variables

x <- data.frame(sapply(data, function(x) data.frame(model.matrix(~ x - 1, data = data))[,-1]))

data\_final <- cbind(x, data[, c("tenure", "MonthlyCharges", "TotalCharges")])

1. **Exploratory Data Analysis (EDA):**

 **Input:** Processed data.

 **Action:**

* Analyze correlations and visualize data distributions.

 **Code:**

# Correlation matrix

nv <- sapply(data\_final, is.numeric)

cormat <- cor(data\_final[, nv])

ggcorrplot::ggcorrplot(cormat, title = "Correlation of Numeric Variables")

# Distribution of churn

churn <- data %>%

group\_by(Churn) %>%

summarise(Count = n()) %>%

mutate(percentage = prop.table(Count) \* 100)

ggplot(churn, aes(reorder(Churn, -percentage), percentage, fill = Churn)) +

geom\_col() +

geom\_text(aes(label = sprintf("%.2f%%", percentage))) +

xlab("Churn") +

ylab("Percent") +

ggtitle("Churn Percentage")

1. **Split Data:**

 **Input:** Processed data.

 **Action:**

* Split dataset into training and testing sets.

 **Code:**

set.seed(123)

split <- sample.split(data\_final$Churn, SplitRatio = 0.70)

train <- data\_final[split, ]

test <- data\_final[!split, ]

1. **Model Development**:

 **Input:** Training data.

 **Action:**

* Train a decision tree model.

 **Code:**

# Train Decision Tree model

rpart\_model <- rpart(Churn ~ ., data = train, method = "class", control = rpart.control(cp = 0.05))

1. **Model Evaluation:**

 **Input:** Testing data.

 **Action:**

* Make predictions and evaluate model performance.

 **Code:**

# Predict and evaluate on test set

rpred\_test <- predict(rpart\_model, newdata = test, type = "class")

# Confusion matrix and accuracy

confusion\_matrix\_test <- table(rpred\_test, test$Churn)

acc\_rpart\_test <- sum(diag(confusion\_matrix\_test)) / nrow(test)

# Print confusion matrix and accuracy

print("Confusion Matrix - Test Set")

print(confusion\_matrix\_test)

print(paste("Accuracy - Test Set:", round(acc\_rpart\_test, 4)))

# Plot the decision tree

rpart.plot(rpart\_model, type = 2, extra = 102, fallen.leaves = TRUE, main = "Decision Tree for Customer Churn")

**Dataset**

 **File:** <https://www.kaggle.com/datasets/blastchar/telco-customer-churn>

 **Description:** Contains historical customer data from a telecommunications company.

 **Columns:**

* customerID (string): Unique identifier for each customer.
* gender (categorical): Gender of the customer.
* SeniorCitizen (binary): Whether the customer is a senior citizen.
* Partner (categorical): Whether the customer has a partner.
* Dependents (categorical): Whether the customer has dependents.
* tenure (numeric): Number of months the customer has been with the company.
* PhoneService (categorical): Whether the customer has phone service.
* MultipleLines (categorical): Whether the customer has multiple lines.
* InternetService (categorical): Type of internet service.
* OnlineSecurity (categorical): Whether the customer has online security.
* OnlineBackup (categorical): Whether the customer has online backup.
* DeviceProtection (categorical): Whether the customer has device protection.
* TechSupport (categorical): Whether the customer has tech support.
* StreamingTV (categorical): Whether the customer has streaming TV.
* StreamingMovies (categorical): Whether the customer has streaming movies.
* Contract (categorical): Type of contract the customer has.
* PaperlessBilling (categorical): Whether the customer has paperless billing.
* PaymentMethod (categorical): Payment method used by the customer.
* MonthlyCharges (numeric): Monthly charges for the customer.
* TotalCharges (numeric): Total charges incurred by the customer.
* Churn (binary): Whether the customer has churned (target variable).