**R Lab for Fundamental Machine Learning Techniques for Classification**

**1. Binary Logistic Regression (Binary Classification Using GLM with Different Link Functions)**

library(readxl)

library(ggplot2)

library(pROC)

**Dataset**

bkdt <- read\_excel("../input/fake-rev/simdata1.xlsx")

**Split data into training (75%) and testing**

set.seed(123)

train\_indices <- sample(1:nrow(bkdt), 0.75 \* nrow(bkdt))

Train <- bkdt[train\_indices, ]

Test <- bkdt[-train\_indices, ]

**Fit models with different link functions**

model\_logit <- glm(LBL ~ RDLTY\_SR, data = Train, family = binomial(link = "logit"))

model\_probit <- glm(LBL ~ RDLTY\_SR, data = Train, family = binomial(link = "probit"))

model\_cloglog <- glm(LBL ~ RDLTY\_SR, data = Train, family = binomial(link = "cloglog"))

**AIC and Summary**

cat("AIC (logit):", AIC(model\_logit), "\n")

summary(model\_logit)

**Predict on test data**

probs <- predict(model\_logit, newdata = Test, type = "response")

predicted <- ifelse(probs > 0.5, 1, 0)

**Confusion Matrix**

cm <- table(Predicted = predicted, Actual = Test$LBL)

print(cm)

**ROC Curve**

roc\_obj <- roc(Test$LBL, probs)

plot(roc\_obj, col = "blue", main = "ROC Curve")

cat("AUC:", auc(roc\_obj), "\n")

**2. Multinomial Logistic Regression (More than 2 classes)(Multinomial Logistic Regression for Mobile Plan Prediction)**

library(nnet)

library(ggplot2)

library(readr)

**Dataset**

data <- read\_csv("../input/mobile-plan/plan.csv")

**Define predictors and response**

X <- data[, c("CSS", "Duration", "MUsage", "Age")]

y <- data$Plan

**Fit multinomial logistic regression**

model <- multinom(Plan ~ CSS + Duration + MUsage + Age, data = data)

**Print coefficients**

print(summary(model))

**Plotting coefficients**

coefs <- coef(model)

features <- colnames(X)

classes <- rownames(coefs)

coef\_df <- data.frame(

Feature = rep(features, each = length(classes)),

Coefficient = as.vector(t(coefs)),

Class = rep(classes, times = ncol(coefs))

)

ggplot(coef\_df, aes(x = Feature, y = Coefficient, color = Class, group = Class)) +

geom\_point() +

geom\_line() +

theme\_minimal() +

ggtitle("Multinomial Logistic Regression Coefficients")

**Predicted probabilities**

probabilities <- predict(model, type = "probs")

head(probabilities)

**Naive Bayes and LDA for Plan Classification**

library(e1071) # for naiveBayes

library(MASS) # for lda

library(caret)

library(readr)

Dataset

data <- read\_csv("plan.csv")

**Naive Bayes: Multiclass**

nb\_data <- data

X <- nb\_data[, c("Age", "MUsage", "CSS", "Duration")]

y <- nb\_data$Plan

train\_index <- createDataPartition(y, p = 0.75, list = FALSE)

train\_data <- nb\_data[train\_index, ]

test\_data <- nb\_data[-train\_index, ]

nb\_model <- naiveBayes(Plan ~ Age + MUsage + CSS + Duration, data = train\_data)

nb\_pred <- predict(nb\_model, newdata = test\_data)

nb\_prob <- predict(nb\_model, newdata = test\_data, type = "raw")

confusionMatrix(nb\_pred, test\_data$Plan)

print("Predicted Probabilities (first 5 rows):")

print(head(nb\_prob))

**LDA: Binary classification (Vanilla vs Pluto)**

binary\_data <- subset(data, Plan %in% c("Vanilla", "Pluto"))

train\_index <- createDataPartition(binary\_data$Plan, p = 0.75, list = FALSE)

train\_lda <- binary\_data[train\_index, ]

test\_lda <- binary\_data[-train\_index, ]

lda\_model <- lda(Plan ~ MUsage + Age, data = train\_lda)

lda\_pred <- predict(lda\_model, newdata = test\_lda)

confusionMatrix(lda\_pred$class, test\_lda$Plan)

print("LDA Coefficients:")

print(lda\_model$scaling)

print("Class Means:")

print(lda\_model$means)