# Lung Cancer Biomarker Analysis

# Set working directory

setwd("C:/Users/vismay/Downloads/biomarker\_analysis")

# 1. Load tumor and normal data

tumor <- read.delim("tumor\_data.csv", sep = "\t", row.names = 1, check.names = FALSE)

normal <- read.delim("normal\_data.csv", sep = "\t", row.names = 1, check.names = FALSE)

# Combine into one count matrix

counts <- cbind(tumor, normal)

dim(counts) # should be (genes, samples)

# Create sample metadata

coldata <- data.frame(

condition = factor(c(rep("Tumor", ncol(tumor)),

rep("Normal", ncol(normal))))

)

rownames(coldata) <- colnames(counts)

# Check sample distribution

table(coldata$condition) # Tumor = 110, Normal = 101

# 2. Differential Expression (DESeq2)

library(DESeq2)

dds <- DESeqDataSetFromMatrix(countData = counts,

colData = coldata,

design = ~ condition)

dds <- DESeq(dds)

res <- results(dds)

res <- res[order(res$padj), ] # sort by adjusted p-value

head(res)

# Save DE results

write.csv(as.data.frame(res), "DE\_results.csv")

# 3. LASSO Feature Selection

library(glmnet)

# Prepare data for glmnet

x <- t(as.matrix(counts)) # samples as rows, genes as columns

y <- factor(coldata$condition, levels = c("Normal", "Tumor"))

y\_bin <- as.numeric(y) - 1 # Normal = 0, Tumor = 1

# Handle missing values

x[is.na(x)] <- 0

for (i in 1:ncol(x)) {

x[is.na(x[, i]), i] <- mean(x[, i], na.rm = TRUE)

}

# Run cross-validated LASSO

set.seed(123)

cvfit <- cv.glmnet(x, y\_bin, family = "binomial", alpha = 1)

# Plot and save CV curve

png("lasso\_cv\_plot.png", width = 800, height = 600)

plot(cvfit)

dev.off()

# Extract selected genes at best lambda

lasso\_genes <- colnames(x)[which(coef(cvfit, s = "lambda.min") != 0)]

lasso\_genes

# Save selected biomarkers

write.csv(lasso\_genes, "lasso\_selected\_biomarkers.csv", row.names = FALSE)

# 4. ROC Curve Validation

library(pROC)

# Predict probabilities

pred\_prob <- predict(cvfit, newx = x, s = "lambda.min", type = "response")

# Compute ROC and AUC

roc\_obj <- roc(y\_bin, as.numeric(pred\_prob))

auc\_value <- auc(roc\_obj)

print(paste("AUC =", auc\_value))

# Plot and save ROC curve

png("roc\_curve.png", width = 800, height = 600)

plot(roc\_obj, col = "blue", lwd = 2, main = "ROC Curve for LASSO Biomarker Model")

abline(a = 0, b = 1, lty = 2, col = "gray"

dev.off()