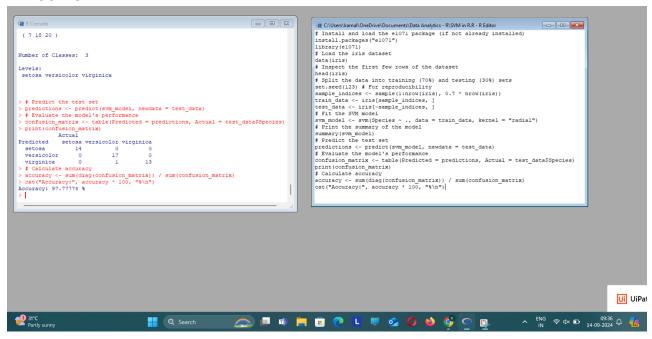
EXP NO: 8 IMPLEMENT SVM/DECISION TREE CLASSIFICATION TECHNIQUES

a) SVM

```
# Install and load the e1071 package (if not already installed)
install.packages("e1071")
library(e1071)
# Load the iris dataset
data(iris)
# Inspect the first few rows of the dataset
head(iris)
# Split the data into training (70%) and testing (30%) sets
set.seed(123) # For reproducibility
sample_indices <- sample(1:nrow(iris), 0.7 * nrow(iris))</pre>
train data <- iris[sample indices, ]
test data <- iris[-sample_indices, ]
# Fit the SVM model
svm model <- svm(Species ~ ., data = train data, kernel = "radial")
# Print the summary of the model
summary(svm model)
# Predict the test set
predictions <- predict(svm model, newdata = test data)</pre>
# Evaluate the model's performance
confusion matrix <- table(Predicted = predictions, Actual = test_data$Species)
print(confusion matrix)
# Calculate accuracy
accuracy <- sum(diag(confusion matrix)) / sum(confusion matrix)
cat("Accuracy:", accuracy * 100, "%\n")
```

OUTPUT:



b) DECISION TREE

```
# Install and load the rpart package (if not already installed)
install.packages("rpart")
library(rpart)
# Load the iris dataset
data(iris)
# Split the data into training (70%) and testing (30%) sets
set.seed(123) # For reproducibility
sample indices <- sample(1:nrow(iris), 0.7 * nrow(iris))
train data <- iris[sample indices,]
test data <- iris[-sample indices, ]
# Fit the Decision Tree model
tree model <- rpart(Species ~ ., data = train data, method = "class")
# Print the summary of the model
summary(tree model)
# Plot the Decision Tree
plot(tree model)
text(tree model, pretty = 0)
# Predict the test set
predictions <- predict(tree model, newdata = test data, type = "class")</pre>
# Evaluate the model's performance
confusion matrix <- table(Predicted = predictions, Actual = test_data$Species)
print(confusion matrix)
# Calculate accuracy
accuracy <- sum(diag(confusion matrix)) / sum(confusion matrix)</pre>
cat("Accuracy:", accuracy * 100, "%\n")
```

OUTPUT:

