

assignment_11_01_PothineniKalyan

PothineniKalyan

2023-05-25

Introduction to machine Learning

Ingest the data and create plots

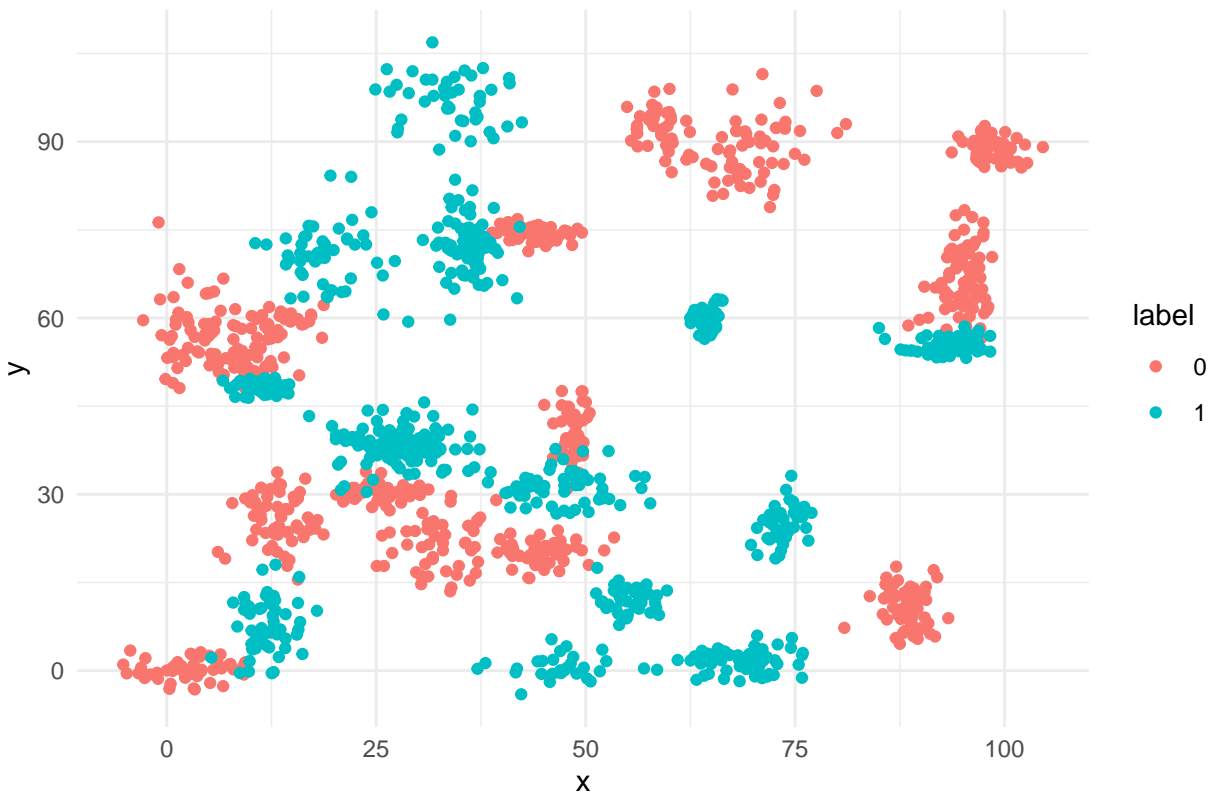
```
# Load required libraries  
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
library(class)
```

```
# Read the binary classifier data  
binary_data <- read.csv("data/binary-classifier-data.csv")  
binary_data$label <- as.factor(binary_data$label)  
  
# Plot the binary classifier data  
ggplot(binary_data, aes(x, y, color = label)) + geom_point() + theme_minimal() +  
  labs(title = "Binary Classifier Data", x = "x", y = "y")
```

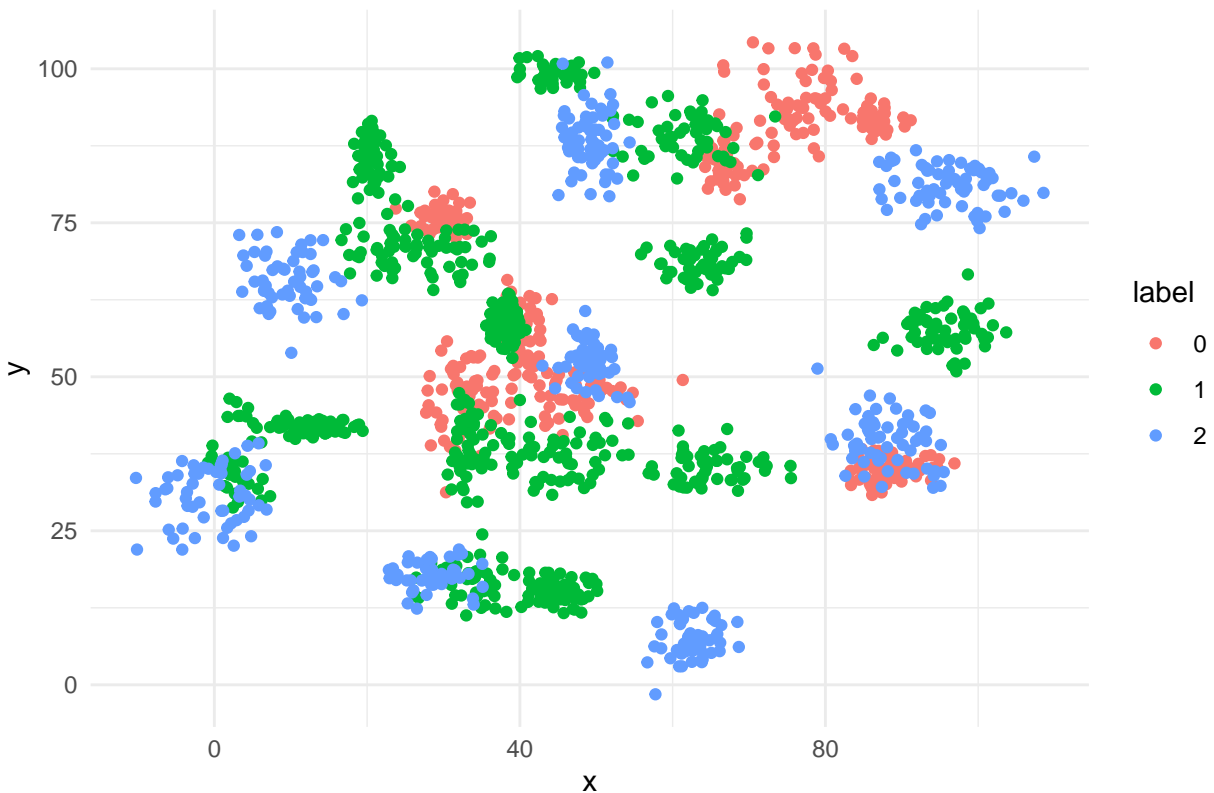
Binary Classifier Data



```
# Read the trinary classifier data
trinary_data <- read.csv("data/trinary-classifier-data.csv")
trinary_data$label <- as.factor(trinary_data$label)

# Plot the trinary classifier data
ggplot(trinary_data, aes(x, y, color = label)) + geom_point() + theme_minimal() +
  labs(title = "Trinary Classifier Data", x = "x", y = "y")
```

Trinary Classifier Data



Euclidean distance calculation

```
# Function to calculate Euclidean distance between two points
euclidean_distance <- function(x1, y1, x2, y2) {
  sqrt((x2 - x1)^2 + (y2 - y1)^2)
}
```

Calculate the nearest neighbor and accuracy

```
# Function to fit k-nearest neighbors model and compute accuracy
fit_knn <- function(data, k) {
  set.seed(123) # For reproducibility

  # Randomly split the data into training and test sets
  train_indices <- sample(1:nrow(data), 0.7*nrow(data))
  train_data <- data[train_indices, ]
  test_data <- data[-train_indices, ]

  # Fit k-nearest neighbors model
  model <- knn(train_data[, c("x", "y")], test_data[, c("x", "y")],
               train_data$label, k)

  # Compute accuracy
  accuracy <- sum(model == test_data$label) / nrow(test_data)

  return(accuracy)
}
```

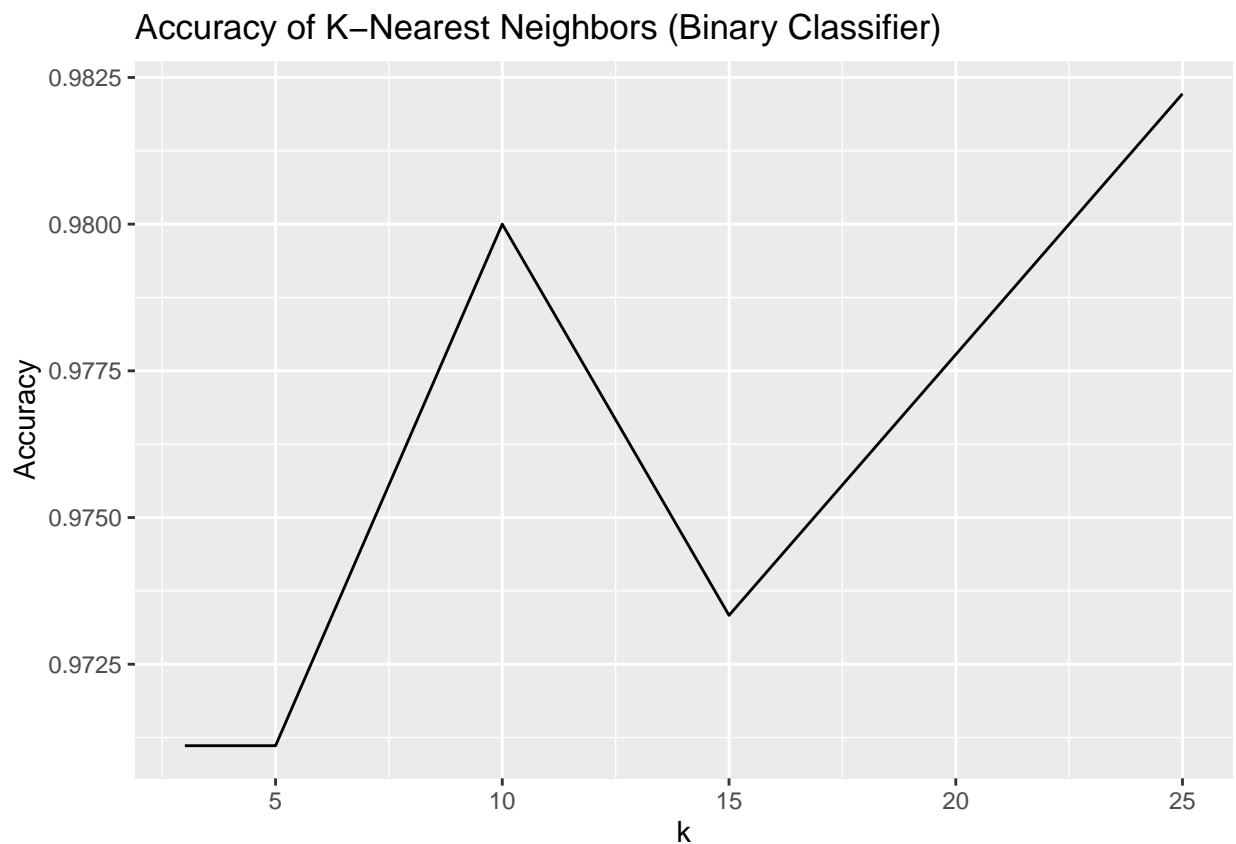
```
}
```

Calculate the accuracies for both datasets and create a data frame to store the the results and gplot to show the accuracies for different k values

```
# Define the values of k
k_values <- c(3, 5, 10, 15, 20, 25)

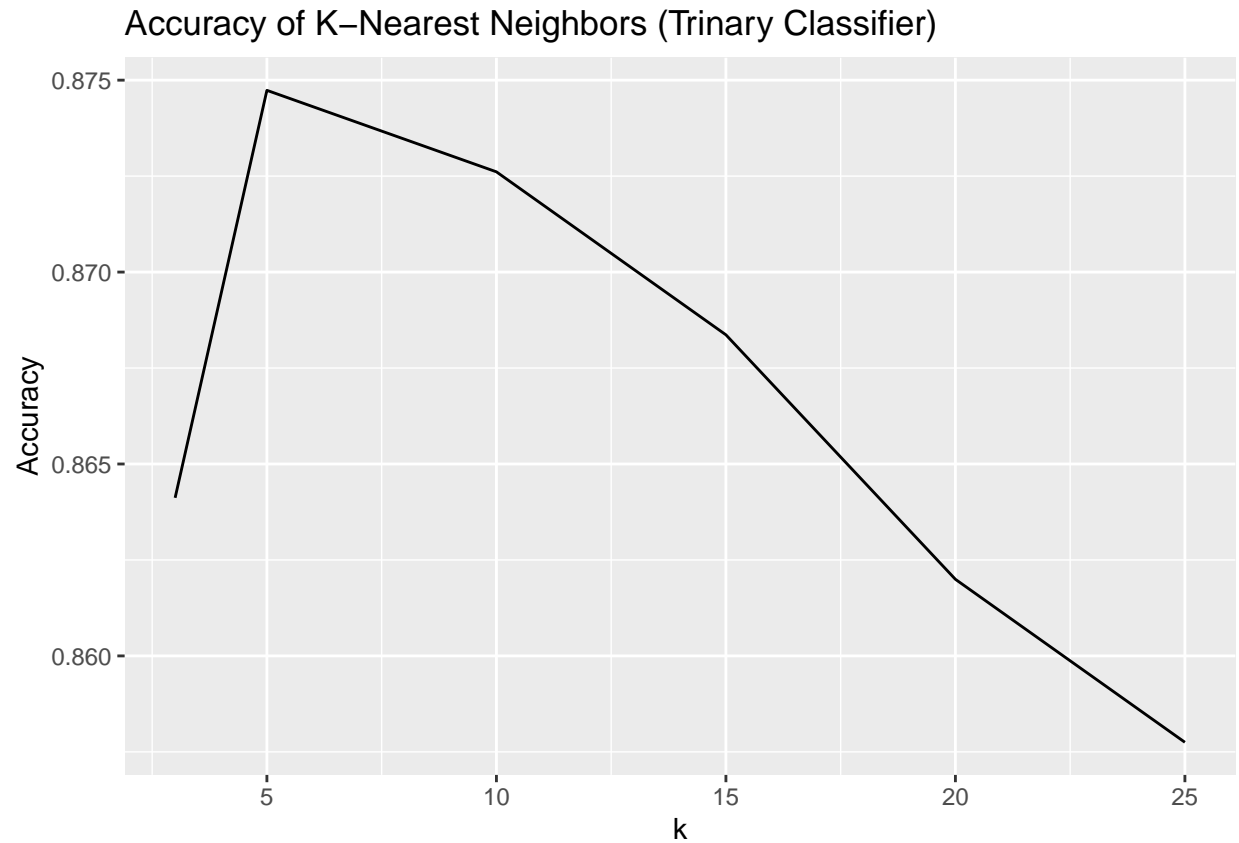
# Fit k-nearest neighbors models for binary classifier dataset
binary_accuracies <- sapply(k_values, function(k) fit_knn(binary_data, k))

# Plot the results for binary classifier dataset
binary_results <- data.frame(k = k_values, accuracy = binary_accuracies)
ggplot(binary_results, aes(k, accuracy)) + geom_line() +
  labs(title = "Accuracy of K-Nearest Neighbors (Binary Classifier)",
       , x = "k", y = "Accuracy")
```



```
# Fit k-nearest neighbors models for trinary classifier dataset
trinary_accuracies <- sapply(k_values, function(k) fit_knn(trinary_data, k))

# Plot the results for trinary classifier dataset
trinary_results <- data.frame(k = k_values, accuracy = trinary_accuracies)
ggplot(trinary_results, aes(k, accuracy)) + geom_line() +
  labs(title = "Accuracy of K-Nearest Neighbors (Trinary Classifier)",
       x = "k", y = "Accuracy")
```



Combining both the accuracies in to single data frame and plotting the plot

```
# combining both the accuracies in to single data frame
combined_results <- data.frame(k = k_values,
                               binary_accuracy = binary_accuracies,
                               trinary_accuracy = trinary_accuracies)

# Print the result
print(combined_results)
```

```
##      k binary_accuracy trinary_accuracy
## 1   3      0.9711111      0.8641189
## 2   5      0.9711111      0.8747346
## 3  10      0.9800000      0.8726115
## 4  15      0.9733333      0.8683652
## 5  20      0.9777778      0.8619958
## 6  25      0.9822222      0.8577495
```

```
# Plot the combined results
ggplot(combined_results, aes(x = k)) +
  geom_line(aes(y = binary_accuracy, color = "Binary Classifier")) +
  geom_line(aes(y = trinary_accuracy, color = "Trinary Classifier")) +
  labs(title = "Accuracy of k Nearest Neighbors Models",
       x = "k",
       y = "Accuracy") +
```

```
scale_color_manual(values = c("blue", "red")) +  
theme_minimal()
```

