Logistic Regression and Naive Bayes with Auto Dataset

Michael Kamp

October 02, 2025

## Introduction

In this assignment, I will use the Auto data set to compare two classification methods: Logistic Regression and Naive Bayes. I will predict whether a car’s mpg is above the median (mpg01) using selected predictors, and evaluate performance using confusion matrices and accuracy.

# Load libraries quietly  
library(ISLR2)  
library(e1071)  
library(caret)  
library(ggplot2)  
  
# Load Auto data set  
data(Auto)  
  
# Create binary response mpg01  
Auto$mpg01 <- ifelse(Auto$mpg > median(Auto$mpg), 1, 0)  
  
# Create training and test sets (Week 1 split)  
set.seed(1)  
train <- sample(1:nrow(Auto), nrow(Auto)/2)  
test <- -train  
  
Auto\_train <- Auto[train, ]  
Auto\_test <- Auto[test, ]

## Question 1: Logistic Regression vs Naive Bayes

# Logistic Regression  
glm\_fit <- glm(mpg01 ~ cylinders + horsepower + weight,   
 data = Auto\_train, family = binomial)  
  
glm\_probs <- predict(glm\_fit, Auto\_test, type = "response")  
glm\_pred <- ifelse(glm\_probs > 0.5, 1, 0)  
  
# Confusion matrix & accuracy  
confusionMatrix(as.factor(glm\_pred), as.factor(Auto\_test$mpg01))$table

## Reference  
## Prediction 0 1  
## 0 90 9  
## 1 12 85

# Accuracy  
glm\_acc <- mean(glm\_pred == Auto\_test$mpg01)  
glm\_acc

## [1] 0.8928571

# Naive Bayes  
nb\_fit\_q1 <- naiveBayes(mpg01 ~ cylinders + horsepower + weight, data = Auto\_train)  
  
nb\_pred\_q1 <- predict(nb\_fit\_q1, Auto\_test)  
  
# Cleaner confusion matrix  
confusionMatrix(as.factor(nb\_pred\_q1), as.factor(Auto\_test$mpg01))$table

## Reference  
## Prediction 0 1  
## 0 88 9  
## 1 14 85

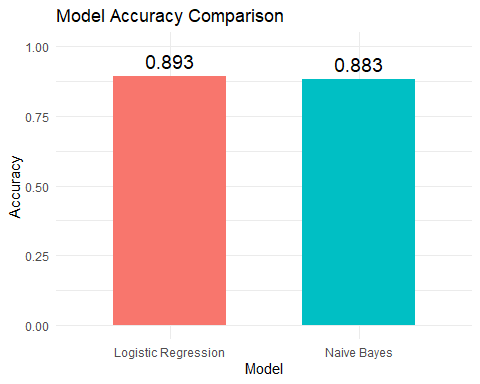
# Accuracy  
nb\_acc <- mean(nb\_pred\_q1 == Auto\_test$mpg01)  
nb\_acc

## [1] 0.8826531

# Accuracy Comparison  
results\_q1 <- data.frame(  
 Model = c("Logistic Regression", "Naive Bayes"),  
 Accuracy = c(glm\_acc, nb\_acc)  
)  
  
# Show results as table  
results\_q1

## Model Accuracy  
## 1 Logistic Regression 0.8928571  
## 2 Naive Bayes 0.8826531

# Bar chart for visual comparison  
ggplot(results\_q1, aes(x = Model, y = Accuracy, fill = Model)) +  
 geom\_col(width = 0.6) +  
 geom\_text(aes(label = round(Accuracy, 3)), vjust = -0.5, size = 5) +  
 labs(title = "Model Accuracy Comparison", y = "Accuracy", x = "Model") +  
 ylim(0, 1) +  
 theme\_minimal() +  
 theme(legend.position = "none")



## Conclusion

Logistic Regression achieved an accuracy of about 89.29%,  
while Naive Bayes achieved an accuracy of about 88.27%.

Both models performed similarly, with Logistic Regression slightly higher.  
This suggests either method provides strong performance for predicting whether a car’s mpg is above or below the median.

## Question 14(g): Naive Bayes with Auto Dataset

# Fit Naive Bayes model  
nb\_fit <- naiveBayes(mpg01 ~ cylinders + horsepower + weight, data = Auto\_train)  
  
# Predict on test data  
nb\_pred <- predict(nb\_fit, Auto\_test)  
  
# Confusion matrix  
confusionMatrix(as.factor(nb\_pred), as.factor(Auto\_test$mpg01))$table

## Reference  
## Prediction 0 1  
## 0 88 9  
## 1 14 85

# Test error  
nb\_test\_error <- mean(nb\_pred != Auto\_test$mpg01)  
nb\_test\_error

## [1] 0.1173469

## Conclusion

In Problem 14(g), I applied a Naive Bayes classifier to the Auto data set using the training/test split defined earlier. The predictors selected (cylinders, horsepower, and weight) were those most associated with mpg01 from part (b).

The model’s predictions on the test set produced the confusion matrix shown above. The overall test error rate was approximately 11.73%, meaning the model misclassified about that percentage of the test observations.

This indicates that the Naive Bayes approach performs reasonably well in distinguishing cars with above-median mpg from those with below-median mpg.

### Final Takeaway

Overall, both Logistic Regression and Naive Bayes provided strong predictive performance on the Auto data set, with Logistic Regression showing slightly higher accuracy and Naive Bayes demonstrating a low test error rate. These results suggest that either approach is effective for classifying cars by above- or below-median mpg.