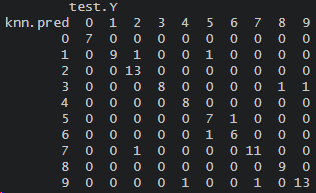
**Problem 1: MNIST handwritten digit database**

1. Optimal K = 5



Misclassification error rate: 0.09

1. Here is the error:  
   A screenshot of a computer code

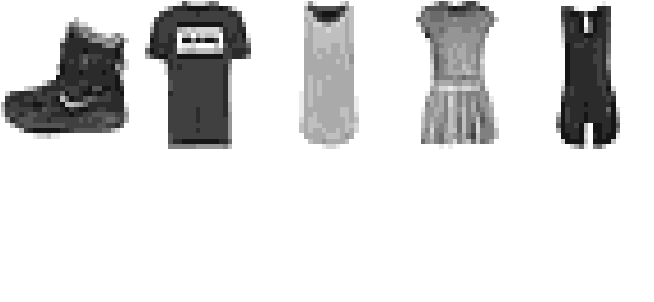
   Description automatically generated

Some limitations of LDA are that if there are substantial separations between the classes, if it is high dimensional data (p>n), irregular decision boundaries, or if there are more than 2 categories. This error seems to be caused because there are more than 2 categories.

1. KNN doesn’t make any assumptions about the data because it is non-parametric, and data driven. It also works well with irregular decision boundaries and when there are more than 2 classification types. KNN works well here where Y can be 0, 1, 2…, 9.

**Problem 2: Fashion MNIST**

1. Plot of the first 5 observations in the training set includes a boot, t-shirt, t-shirt, dress, t-shirt



1. Optimal K=5

A screenshot of a computer screen

Description automatically generated

Misclassification error rate: 0.25

Compared to 1a, this misclassification is larger so the model in 1a has better results.

**Problem 3: Concept Review**

1. Using KNN for a classification when there is a large number of predictors relative to the sample size can lead to several challenges. In such cases, the curse of dimensionality becomes a significant issue. With 10,000 predictors and 100,000 observations, the feature space is highly sparse, making it challenging to identify meaningful neighbors in the high-dimensional space. This leads to a greater likelihood of finding neighbors that are not representative of the true underlying patterns, leading to poor classification performance. Additionally, the computational cost of calculating distances between data points in high-dimensional space can be prohibitive. KNN’s effectiveness relies on having a sufficient amount of data to accurately capture the local structure of the feature space, which can be compromised when p is large relative to the sample size. So KNN might fail in this scenario.
2. (i) I would suggest using logistic regression. This is a binary classification problem where the goal is to predict one of two possible outcomes (male or female). Logistic regression would work well when the relationship between the predictors and the response is expected to be somewhat linear. In this scenario, there is a relatively small dataset, so logistic regression is computationally efficient and can provide meaningful insights.

(ii) I would suggest using logistic regression or LDA. Both models are suitable for binary classification and work well with continuous predictors. The choice between logistic regression and LDA may come down to the specific characteristics of the data and the assumptions of each method. You can try both and see which provides a better performance based on their confusion matrices and misclassification error rates and cross-validation.

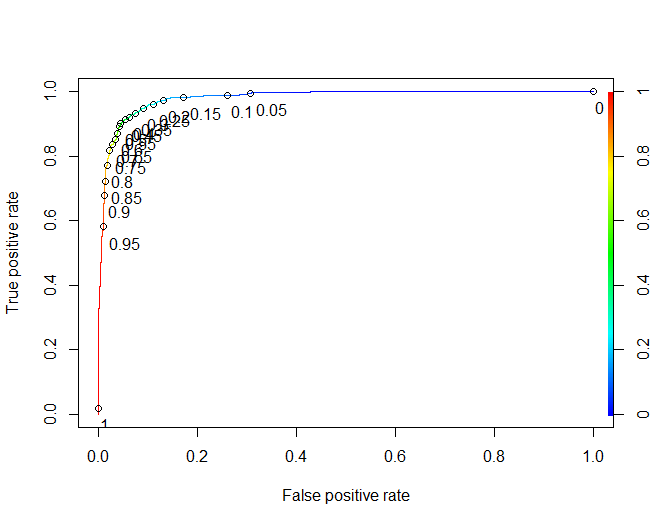
(iii) For predicting gender when the decision boundary is complicated and highly non-linear, with a training set of 960 men and 1040 women, I would suggest using KNN. It is well suited for problems with complex and non-linear decision boundaries because it doesn’t make strong parametric assumptions about the data. By considering the nearest neighbors in the feature space, KNN can capture intricate decision boundaries. KNN is more flexible and can accommodate the non-linearity that can be expected in this classification problem.

**Problem 4: k-NN**

1. To classify the test observation using K=1, choose the observation from the training set with the shortest distance to the test observation. Here, the nearest neighbor is training observation 2 with a distance of 0.23 and it’s corresponding Y­i label is 1. The test observation would be classified as Y=1 when K=1.
2. To classify the test observation using K=3: (1) Training Observation 2 with a distance of 0.23 and Yi = 1. (2) Training Observation 4 with a distance of 0.31 and Yi = 0. (3) Training Observation 1 with a distance of 0.45 and Yi = 0. The test observation would be classified as Y=0 when K=3.
3. When K is small, the model has low bias, meaning it can capture complex and intricate patterns in the data. But this can lead to high variance because it becomes sensitive to noise and outliers in the training data, resulting in a less stable prediction. When K is large, the model has high bias which leads to lower variance, but it may result in oversimplification and a potential loss of important details in the data.

**Problem 5: Email Spam Part 2**

1. False positives seem highly problematic. I do not want a potentially important email to be marked as spam. Therefore, I can tune the threshold for logistic regression that my spam filter is more conservative and makes it harder to mark emails as spam.
2. Here is the ROC curve:



1. Confusion matrix:



False positive rate: 0.06180556

False negative rate: 0.07433217

1. Threshold of 0.15

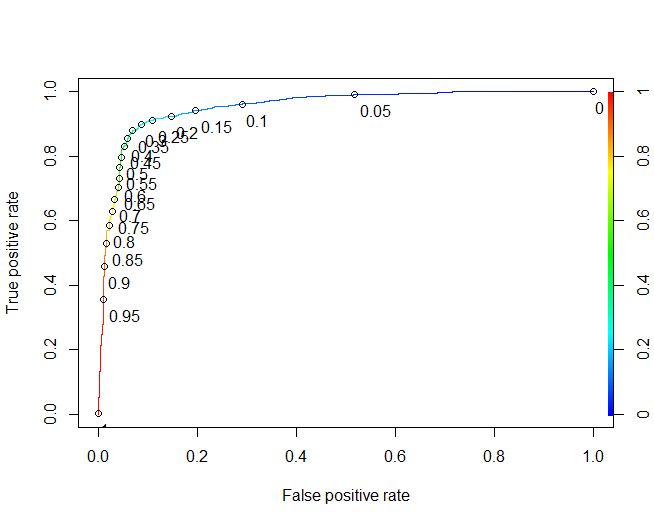
A number on a black background

Description automatically generated

False positive rate: 0.01346801

False negative rate: 0.2183288

1. Here is the ROC curve:



Confusion matrix with a threshold of 0.5 is:  
A black background with white text

Description automatically generated  
False positive rate: 0.04664311

False negative rate: 0.2031603

Confusion matrix with a threshold of 0.08 is:  
A black background with white text

Description automatically generated

False positive rate: 0.02651113

False negative rate: 0.3659794

1. QDA Confusion matrix with a threshold of 0.08 is:

A black background with white text

Description automatically generated  
False positive rate: 0.235963

False negative rate: 0.04697987

Naïve Bayes Confusion matrix:  


False positive rate: 0.05140187

False negative rate: 0.4117647

KNN Confusion matrix:

A black background with white text

Description automatically generated

False positive rate: 0.09190372

False negative rate: 0.1741935

1. I would recommend using LDA because it can be valuable for identifying and classifying spam emails versus non-spam emails. LDA assumes that the predictors are normally distributed which can be a reasonable assumption for many types of email-related features. The threshold that was used also produced the best false positive error rate compared to the other models.