DATA 558: HW Assignment 1

Hriday Baghar

April 12, 2022

1. Suppose that you are interested in performing regression on a particular dataset, in order to answer a particular scientific question. You need to decide whether to take a parametric or a non-parametric approach.

(a) In general, what are the pros and cons of taking a parametric versus a non-parametric approach?

	Parametric model	Non-parametric model
Pros	- Simplifies the problem from estimating a high-dimensional function to represent f to a set of parameters - Interpretability of the model is easier using parameters	- Does not assume a functional form of f, making it easier to fit a wider range of possible shapes for f - Performs well when statistical inference is not a concern, flexibility is high
Cons	- The model may not be a close estimate of the true form of f. Hence if it is too far off from the true f, our estimate will be poor	- Requires a large number of observations in comparison to the number of features - While flexibility is high, interpretability is low for these models

(b) What properties of the data or scientific question would lead you to take a parametric approach?

1. Requirement of model interpretability: If the scientific question requires the ability to explain the relation between the response and predictors and to make statistical inferences, we should prefer parametric methods
2. Limited number of data points: If we do not have a higher number of observations, parametric methods may perform better than non-parametric methods

(c) What properties of the data or scientific question would lead you to take a non-parametric approach?

1. High number of data points: If we have a much larger number of observations than features in the data, we can use parametric methods
2. Requirement of model accuracy: If we are more concerned about model accuracy and do not care for interpretability, non-parametric methods can create complex models that closely estimate the true f.

2. In each setting, would you generally expect a flexible or an inflexible statistical machine learning method to perform better? Justify your answer.

(a) Sample size n is very small, and number of predictors p is very large.

Inflexible will be better. If we use a flexible model, it will overfit the training data and hence perform worse than an inflexible model on the test data.

(b) Sample size n is very large, and number of predictors p is very small.

Flexible will be better. A flexible model with a large number of observations will better estimate f and the large number of observations will allow the model to generalize better for unknown data, provided that the training data is representative of the data distribution.

(c) Relationship between predictors and response is highly non-linear.

Flexible will be better. A flexible model will be able to create a better estimate of the non-linearity between the response and the predictors. Inflexible methods will constrain the shapes that the estimated model can take on thereby not being able to represent this non-linearity as well as a flexible model.

(d) The variance of the error terms, i.e. $\sigma^2 = Var(\epsilon)$, is extremely high.

Inflexible will be better. If the variance of the error terms is high, that means we are running into an overfitting problem, i.e. the model is a good estimator for the training data but not so much for the test data. Using an inflexible approach will help the model better generalize the underlying relationship in the test set.

3. For each scenario, determine whether it is a regression or a classification problem, determine whether the goal is inference or prediction, and state the values of n (sample size) and p (number of predictors).

(a) I want to predict each student's final exam score based on their homework scores. There are 50 students enrolled in the course, and each student has completed 8 homeworks.

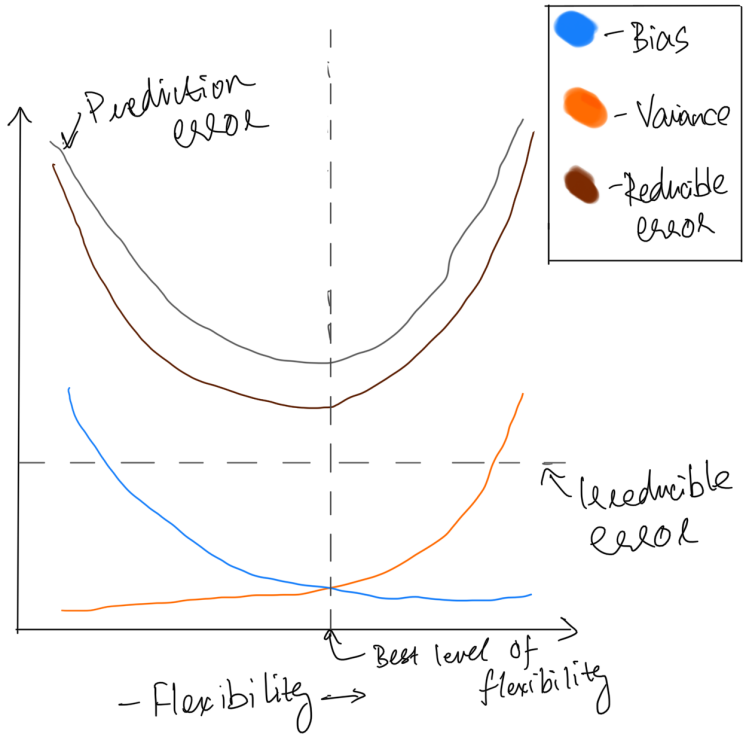
- *Problem:* Regression
- *Goal:* Prediction
- *n:* 50
- *p:* 8

(b) I want to understand the factors that contribute to whether or not a student passes this course. The factors that I consider are (i) whether or not the student has previous programming experience; (ii) whether or not the student has previously studied linear algebra; (iii) whether or not the student has taken a previous stats/probability course; (iv) whether or not the student attends office hours; (v) the student's overall GPA; (vi) the student's year (e.g. freshman, sophomore, junior, senior, or grad student). I have data for all 50 students enrolled in the course.

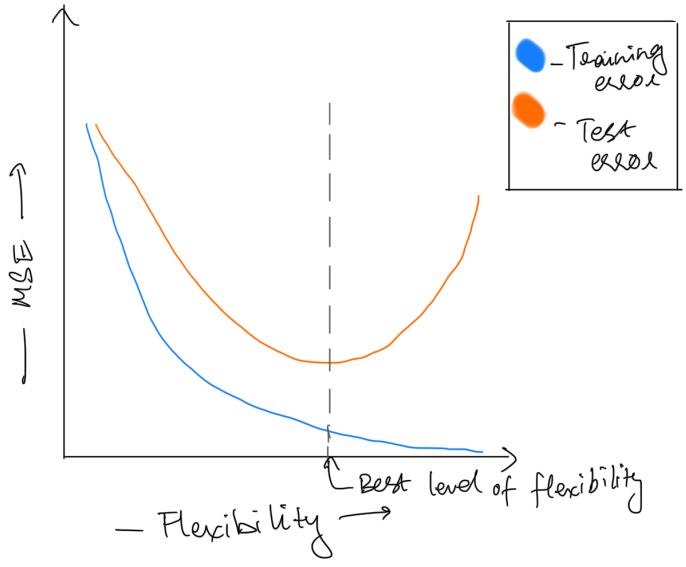
- *Problem:* Classification
- *Goal:* Inference
- *n:* 50
- *p:* 6

4. This problem has to do with the bias-variance trade-off and related ideas, in the context of regression. For (a) and (b), it's okay to submit hand-sketched plots: you are not supposed to actually compute the quantities referred to below on data; instead, this is a thought exercise.

(a) Make a plot, like the one we saw in class, with "flexibility" on the x-axis. Sketch the following curves: squared bias, variance, irreducible error, reducible error, expected prediction error. Be sure to label each curve. Indicate which level of flexibility is "best".



(b) Make a plot with "flexibility" on the x-axis. Sketch curves corresponding to the training error and the test error. Be sure to label each curve. Indicate which level of flexibility is "best".



(c) Describe an $\hat{f}$ that has extremely low bias, and extremely high variance. Explain your answer.

An $\hat{f}$ that passes very closely through each point in the training data will have extremely low bias and extremely high variance. This is because the bias is the difference between the true value and estimated value (which in this case will be low for the model we describe). The variance will be high because for a different training set, the $\hat{f}$ will look very different, as it depends highly on what training data is observed.

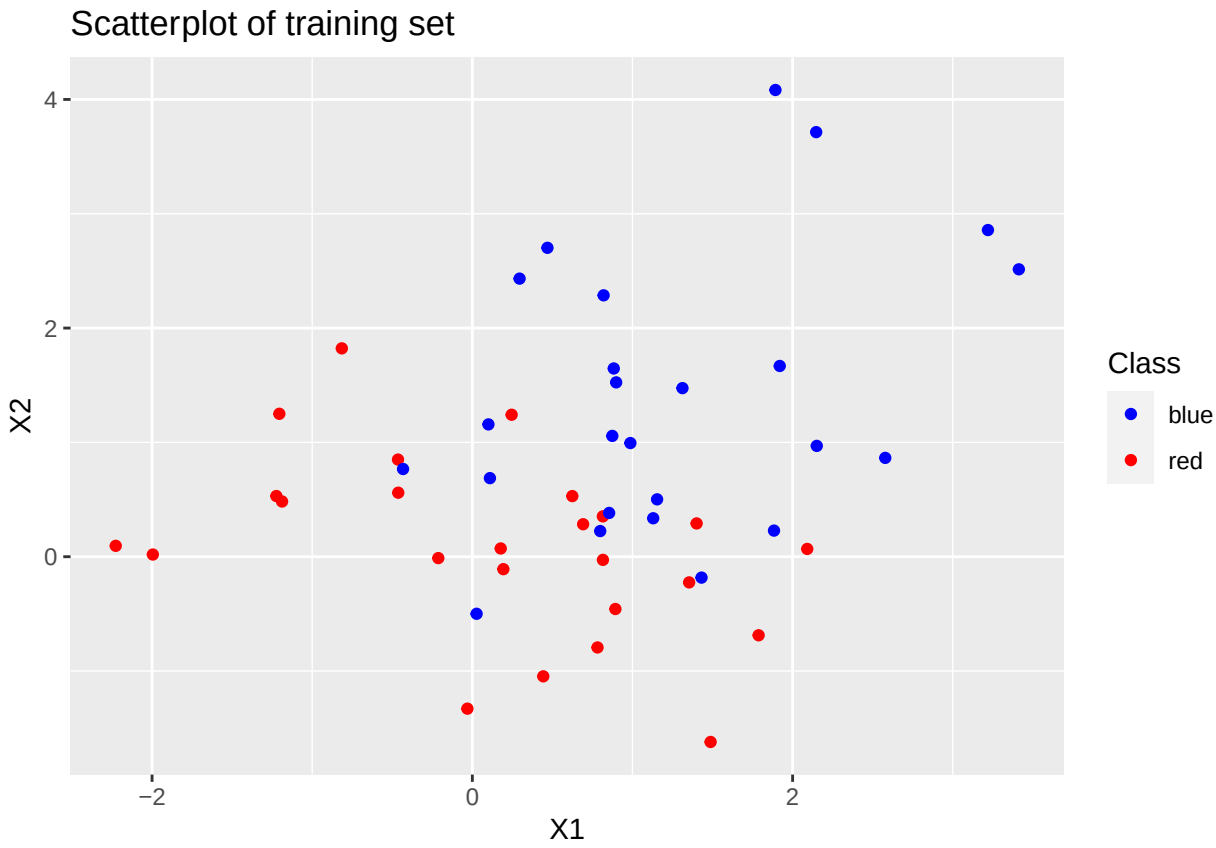
(d) Describe an $\hat{f}$ that has extremely high bias, and zero variance. Explain your answer.

A model that only has a randomly selected intercept value would have extremely high bias and zero variance. Bias would be extremely high because the model will have estimates that are way off from the true values. Variance will be zero because the model always returns a single value for $\hat{f}(X_0)$ no matter what the value of X_0 is. Hence, $Var(\hat{f}(X_0)) = 0$.

5. We now consider a classification problem. Suppose we have 2 classes (labels), 25 observations per class, and p = 2 features. We will call one class the "red" class and the other class the "blue" class. The observations in the red class are drawn i.i.d. from a $N_p(\mu_r, I)$ distribution, and the observations in the blue class are drawn i.i.d. from a $N_p(\mu_b, I)$ distribution, where $\mu_r = \binom{0}{0}$ is the mean in the red class, and where $\mu_b = \binom{1.5}{1.5}$ is the mean in the blue class.

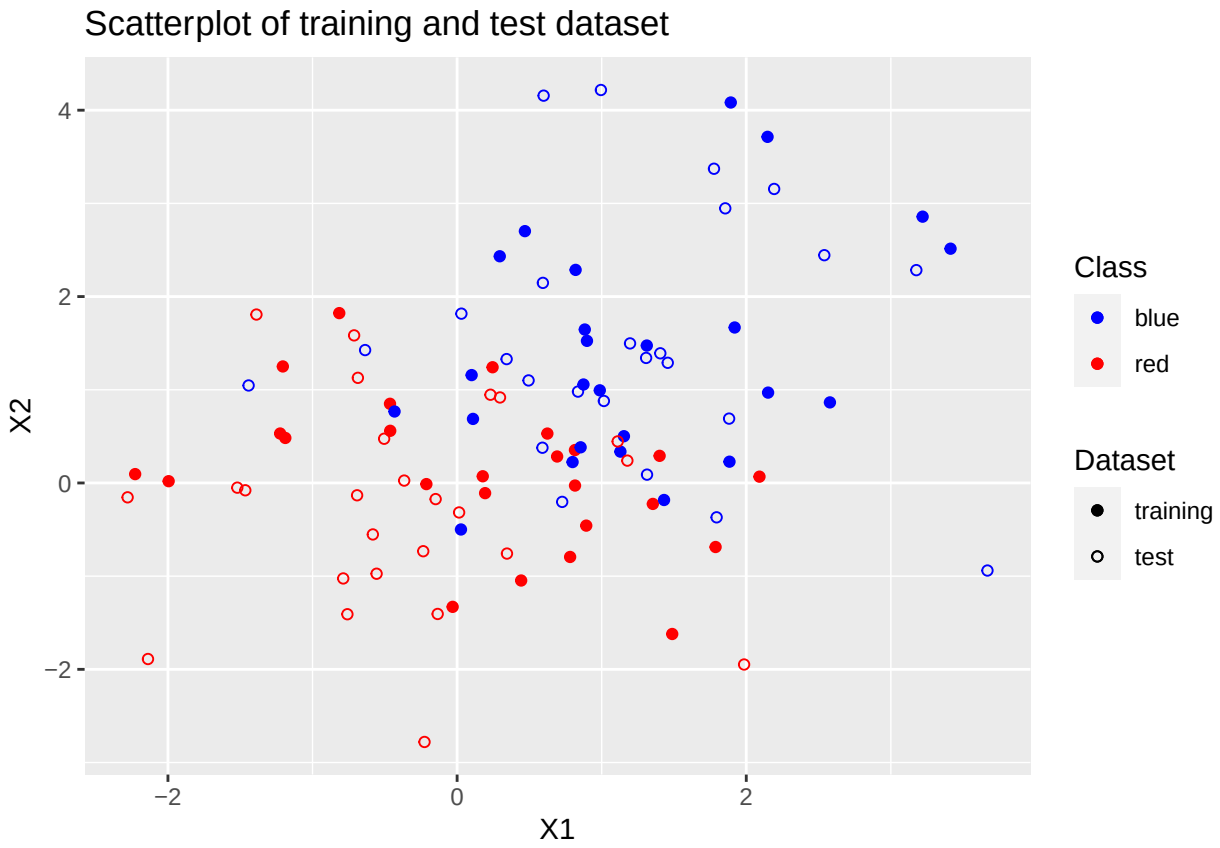
(a) Generate a training set, consisting of 25 observations from the red class and 25 observations from the blue class. (You will want to use the R function rnorm.) Plot the training set. Make sure that the axes are properly labeled, and that the observations are colored according to their class label.

```
set.seed(558)

x.train.red <- matrix(rnorm(50), nrow = 25, ncol = 2)
x.train.blue <- matrix(rnorm(50, mean = 1.5), nrow = 25, ncol = 2)

x.train <- data.frame(rbind(x.train.red, x.train.blue),
                      class = c(rep("red",25),rep("blue", 25)),
                      stringsAsFactors = TRUE)

library(ggplot2)

color.names <- levels(x.train$class)
names(color.names) <- color.names
color.scale.gg <- scale_color_manual(name = "Class", values=color.names)

ggplot(data = x.train, aes(x=X1, y=X2, color=class)) +
  geom_point() +
  color.scale.gg +
  ggtitle("Scatterplot of training set")
```



(b) Now generate a test set consisting of 25 observations from the red class and 25 observations from the blue class. On a single plot, display both the 2 training and test set, using one symbol to indicate training observations (e.g. circles) and another symbol to indicate the test observations (e.g. squares). Make sure that the axes are properly labeled, that the symbols for training and test observations are explained in a legend, and that the observations are colored according to their class label.

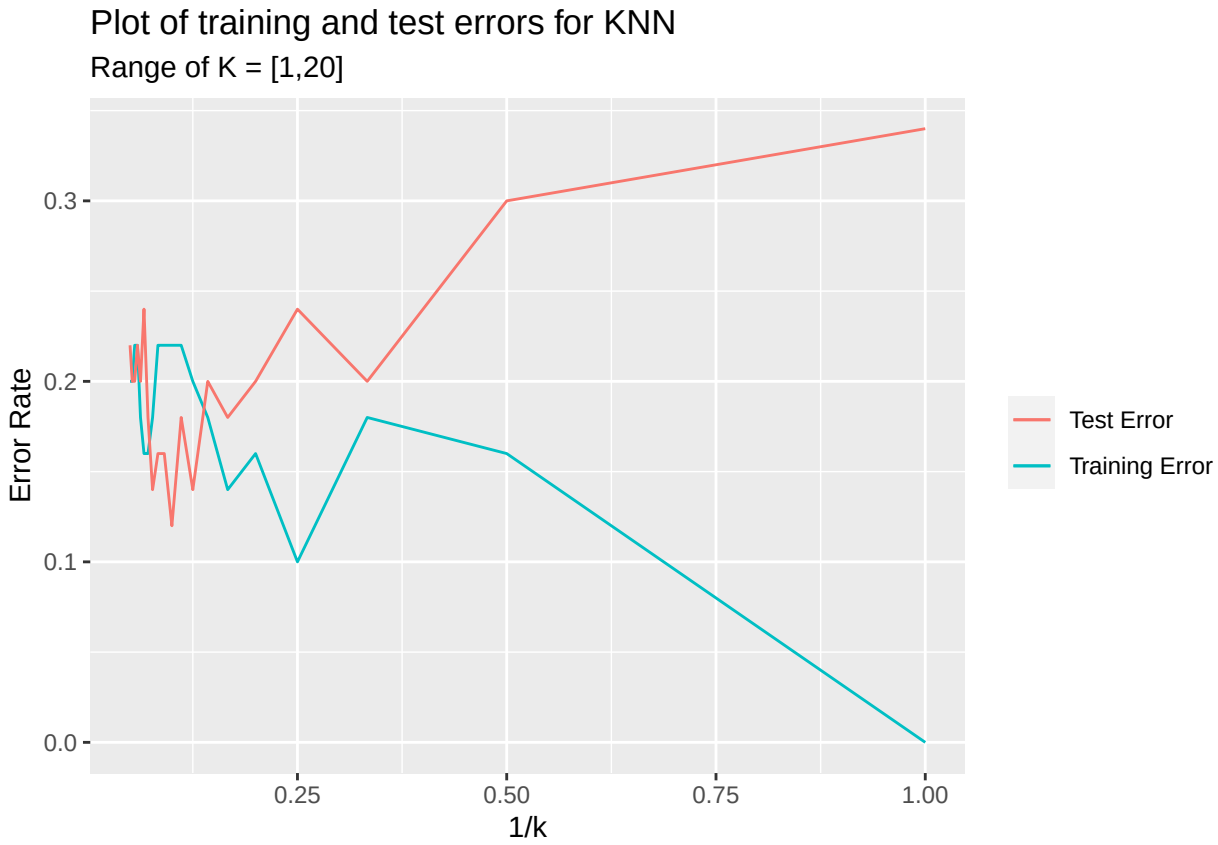
```
x.test.red <- matrix(rnorm(50), nrow = 25, ncol = 2)
x.test.blue <- matrix(rnorm(50, mean = 1.5), nrow = 25, ncol = 2)

x.test <- data.frame(rbind(x.test.red, x.test.blue),
                     class = c(rep("red",25), rep("blue", 25)),
                     stringsAsFactors = TRUE)

shape.scale.gg <- scale_shape_manual("Dataset", values = c("training"=19, "test"=1))

ggplot() +
  geom_point(data = x.train, aes(x=X1, y=X2, color = class, shape = "training")) +
  geom_point(data = x.test, aes(x=X1, y=X2, color = class,  shape = "test")) +
  color.scale.gg +
  shape.scale.gg +
  ggtitle("Scatterplot of training and test dataset")
```



(c) Using the knn function in the library class, fit a k-nearest neighbors model on the training set, for a range of values of k from 1 to 20. Make a plot that displays the value of 1/k on the x-axis, and classification error (both training error and test error) on the y-axis. Make sure all axes and curves are properly labeled. Explain your results.

```
library(class)

K <- seq(1,20)

y.train <- x.train[, c(3)]
x.train <- x.train[,-c(3)]

y.test <- x.test[, c(3)]
x.test <- x.test[, -c(3)]

compute.knn.metrics <- function(k){
  train.knn <- knn(train = x.train, test = x.train, cl = y.train, k = k)
  test.knn <- knn(train = x.train, test = x.test, cl = y.train, k = k)

  train.err <- 1 - length(which(train.knn == y.train))/nrow(x.train)
  test.err <- 1 - length(which(test.knn == y.test))/nrow(x.test)

  return(c(train.err, test.err, k, test.knn))
}

results <- t(sapply(K, function(k) compute.knn.metrics(k)))
```

```
error.results <- data.frame(results[,c(1:3)])
names(error.results) <- c("training.error", "test.error", "k")

ggplot(data = error.results, aes(x = 1/k)) +
  geom_line(aes(y=training.error, color="Training Error")) +
  geom_line(aes(y=test.error, color="Test Error")) +
  labs(y = "Error Rate", title = "Plot of training and test errors for KNN",
       subtitle = "Range of K = [1,20]") +
  theme(legend.title = element_blank())
```



We see from the graph that, as k decreases (1/k increases), the training error drops eventually to 0, however the corresponding test error increases. k = 1 is an example of a highly flexible model, and the graph depicts the problem with more flexibility - the tendency to overfit training data.

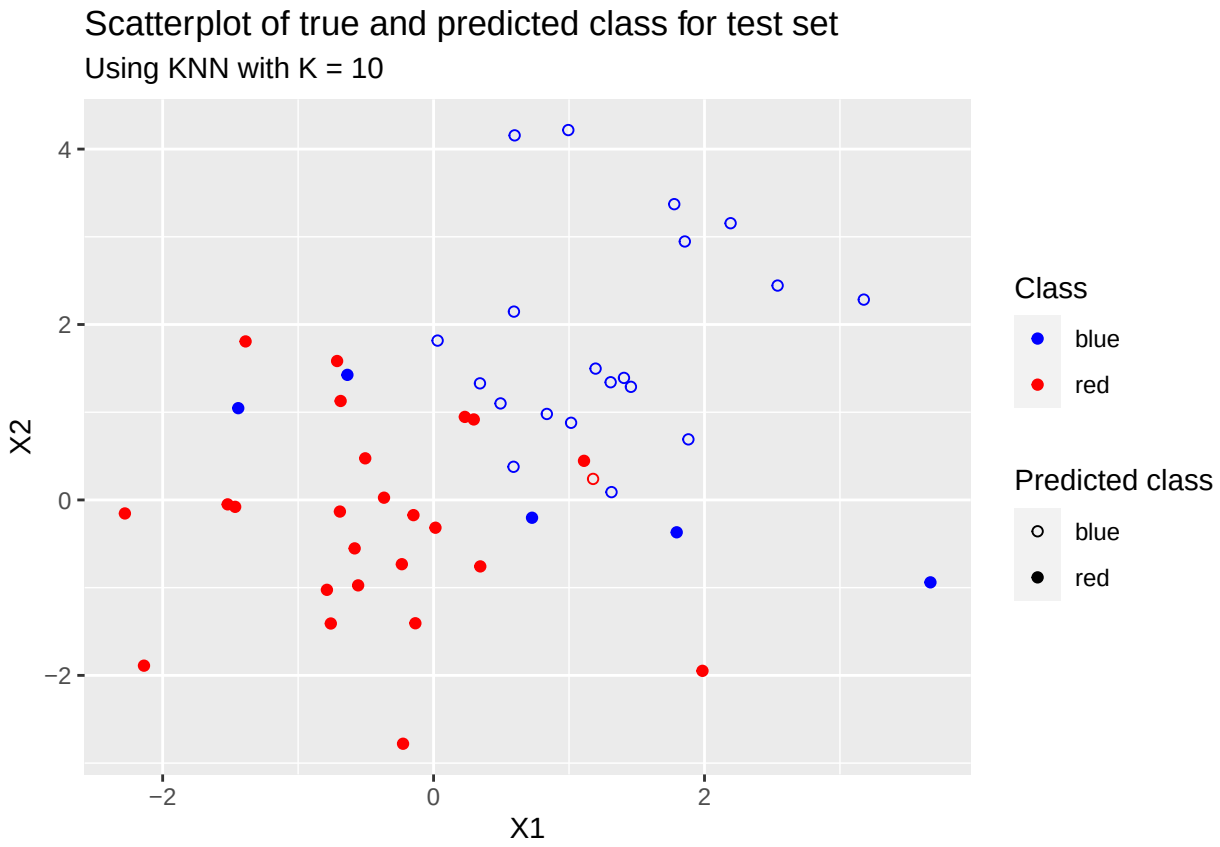
This can also be a sign that a highly flexible or non-linear model is not as good a fit than less flexible models, as we see a steadily rising test error as the value of k decreases.

(d) For the value of k that resulted in the smallest test error in part (c) above, make a plot displaying the test observations as well as their true and predicted class labels. Make sure that all axes and points are clearly labeled.

```
min.k <- which(error.results$test.error == min(error.results$test.error))
#Above value turns out to be 10
print(min.k)
```

```
## [1] 10
```

```
predictions <- as.factor(results[min.k, -c(1:3)])
levels(predictions) <- levels(y.test)
best.test.result <- data.frame(x.test, y.test, predictions)

shape.scale.gg <- scale_shape_manual("Predicted class",
                                     values = c("blue"=1,"red"=19))

ggplot(data = best.test.result, aes(x = X1, y = X2, color = y.test,
                                    shape = predictions)) +
  geom_point() +
  color.scale.gg +
  shape.scale.gg +
  ggtitle("Scatterplot of true and predicted class for test set",
          subtitle = "Using KNN with K = 10")
```



(e) Recall that the Bayes classifier assigns an observation to the red class if $Pr(Y = red|X = x) > 0.5$, and to the blue class otherwise. The Bayes error rate is the error rate associated with the Bayes classifier. What is the value of the Bayes error rate in this problem? Explain your answer.

We will calculate the euclidean distance of each point from the mean of the two distributions i.e. $\mu_r = \binom{0}{0}$ and $\mu_b = \binom{1.5}{1.5}$ and assign the class to the observation whose mean is closest to it. We do this because the $P(Y = red|X) > 0.5$ for a point that is closer to $\mu_r = \binom{0}{0}$ and the same for Y=blue and μ_b.

We can make the above assumption because both distributions are symmetric normal distributions with equal variances,hence the decision boundary is a straight line equidistant from both clusters.

We simulate a large number of test observations from the two distributions and apply the above described process to calculate the Bayes error rate.

```
x.sim.red <- matrix(rnorm(1e6), nrow = 0.5e6, ncol = 2)
x.sim.blue <- matrix(rnorm(1e6, mean = 1.5), nrow = 0.5e6, ncol = 2)

x.sim <- data.frame(rbind(x.sim.red, x.sim.blue),
                    class = c(rep("red",0.5e6), rep("blue", 0.5e6)),
                    stringsAsFactors = TRUE)

bayes.classifer <- function(x) {
  x$dist.red <- sqrt((x.sim[,1] - 0)^2 + (x.sim[,2] - 0)^2)
  x$dist.blue <- sqrt((x.sim[,1] - 1.5)^2 + (x.sim[,2] - 1.5)^2)

  x$predicted <- ifelse(x$dist.red < x$dist.blue, "red", "blue")
  levels(x$predicted) <- levels(x$class)

  err <- 1-length(which(x$class == x$predicted))/nrow(x)

  return(err)
}

bayes.error <- bayes.classifer(x.sim)
cat("Error rate of Bayes classifier = ", bayes.error)
```

```
## Error rate of Bayes classifier =  0.14416
```

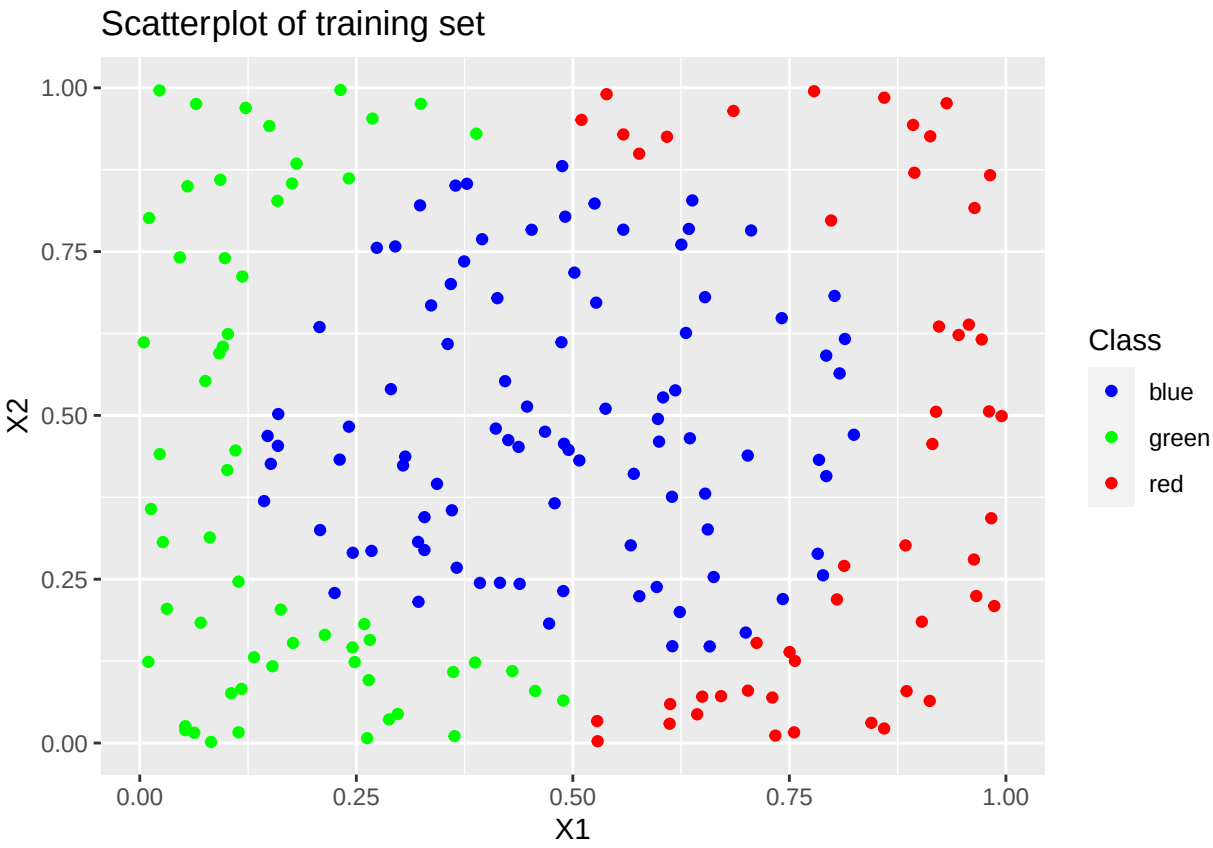
6. We will once again perform k-nearest-neighbors in a setting with p = 2 features. But this time, we'll generate the data differently: let $X_1 \sim Unif[0,1]$ and $X_2 \sim Unif[0,1]$, i.e. the observations for each feature are i.i.d. from a uniform distribution. An observation belongs to class "red" if $(X_1-0.5)^2+(X_2-0.5)^2 > 0.15$ and $X_1 > 0.5$; to class "green" if $(X_1-0.5)^2+(X_2-0.5)^2 > 0.15$ and $X_1 \leq 0.5$; and to class "blue" otherwise.

```
assign.class <- function(x1,x2) {
  if(((x1-0.5)^2 + (x2-0.5)^2) > 0.15 && x1 > 0.5){
    return("red")
  }
  else if(((x1-0.5)^2 + (x2-0.5)^2) > 0.15 && x1 <= 0.5){
    return("green")
  }
  else{
    return("blue")
  }
}
```

(a) Generate a training set of n = 200 observations. (You will want to use the R function runif.) Plot the training set. Make sure that the axes are properly labeled, and that the observations are colored according to their class label.

```
set.seed(558)
```

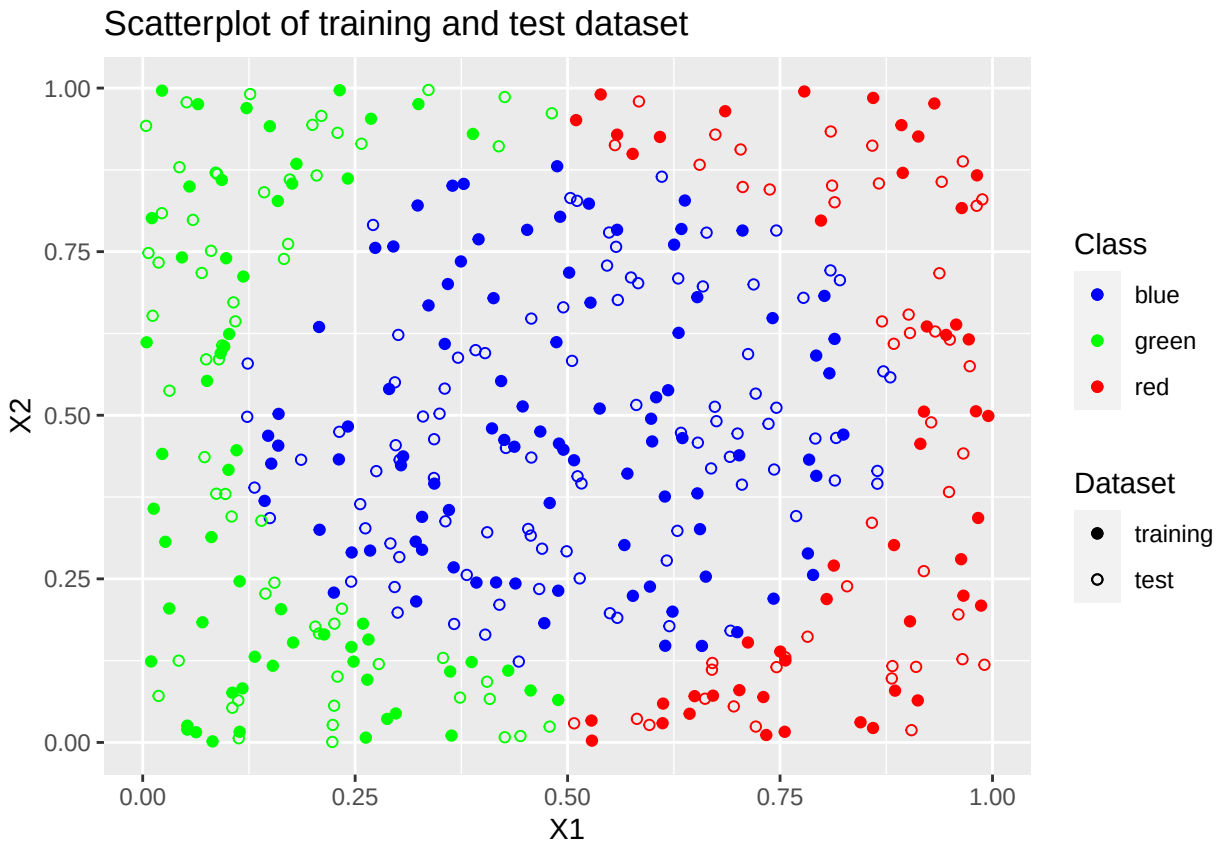
```
x.mat <- matrix(runif(400), ncol = 2)
y.train <- as.factor(apply(x.mat, MARGIN = 1, function(x) assign.class(x[1], x[2])))

x.train <- data.frame(x.mat, class = y.train)

color.names <- levels(x.train$class)
names(color.names) <- color.names
color.scale.gg <- scale_color_manual(name = "Class", values=color.names)

ggplot(data = x.train, aes(x=X1, y=X2, color=class)) +
  geom_point() +
  color.scale.gg +
  ggtitle("Scatterplot of training set")
```



(b) Now generate a test set consisting of another 200 observations. On a single plot, display both the training and test set, using one symbol to indicate training observations (e.g. circles) and another symbol to indicate the test observations (e.g. squares). Make sure that the axes are properly labeled, that the symbols for training and test observations are explained in a legend, and that the observations are colored according to their class label.

```
x.mat <- matrix(runif(400), ncol = 2)
y.test <- as.factor(apply(x.mat, MARGIN = 1, function(x) assign.class(x[1], x[2])))

levels(y.test) <- levels(y.train)

x.test <- data.frame(x.mat, class = y.test)
```

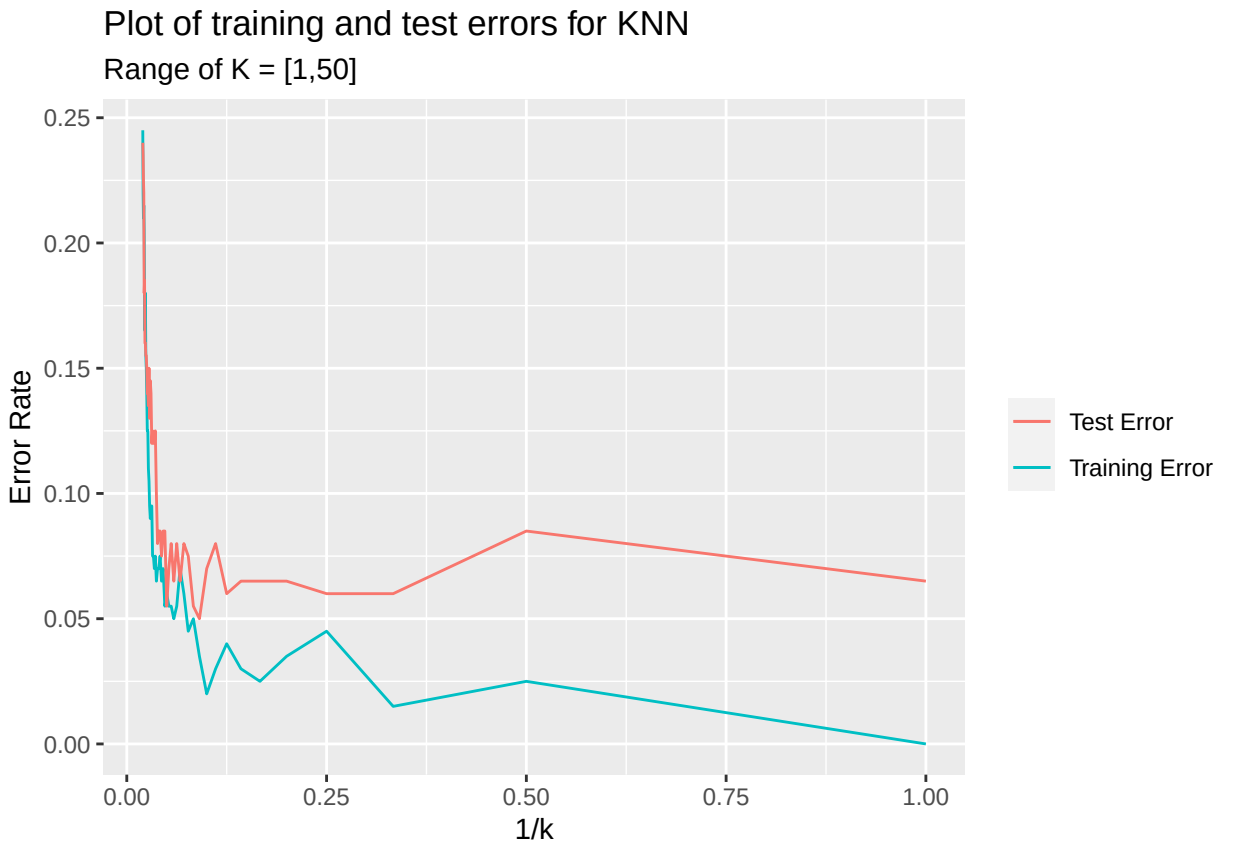
```
shape.scale.gg <- scale_shape_manual("Dataset", values = c("training"=19, "test"=1))

ggplot() +
  geom_point(data = x.train, aes(x=X1, y=X2, color = class, shape = "training")) +
  geom_point(data = x.test, aes(x=X1, y=X2, color = class,  shape = "test")) +
  color.scale.gg +
  shape.scale.gg +
  ggtitle("Scatterplot of training and test dataset")
```



(c) Using the knn function in the library class, fit a k-nearest neighbors model on the training set, for a range of values of k from 1 to 50. Make a plot that displays the value of 1/k on the x-axis, and classification error (both training error and test error) on the y-axis. Make sure all axes and curves are properly labeled. Explain your results.

```
K <- seq(1,50)

x.train <- x.train[,-c(3)]
x.test <- x.test[,-c(3)]

results <- t(sapply(K, function(k) compute.knn.metrics(k)))
error.results <- data.frame(results[,c(1:3)])
names(error.results) <- c("training.error", "test.error", "k")

ggplot(data = error.results, aes(x = 1/k)) +
  geom_line(aes(y=training.error, color="Training Error")) +
```

```
  geom_line(aes(y=test.error, color="Test Error")) +
  labs(y = "Error Rate", title = "Plot of training and test errors for KNN",
       subtitle = "Range of K = [1,50]") +
  theme(legend.title = element_blank())
```



We see that for larger values of k (lower 1/k), the training and test error are very high. As the value of k decreases, model flexibility increases and such models are a better fit for the data. We can guess from the scatter plot that the decision boundary is likely to be non-linear and this is reflected in the above graph too as more flexible models (lower k) have lower test error than less flexible models (higher k).

The reason why the smallest value of k is still not the best model is because of overfitting. The model is highly tuned for the training data but not so much for the test data.

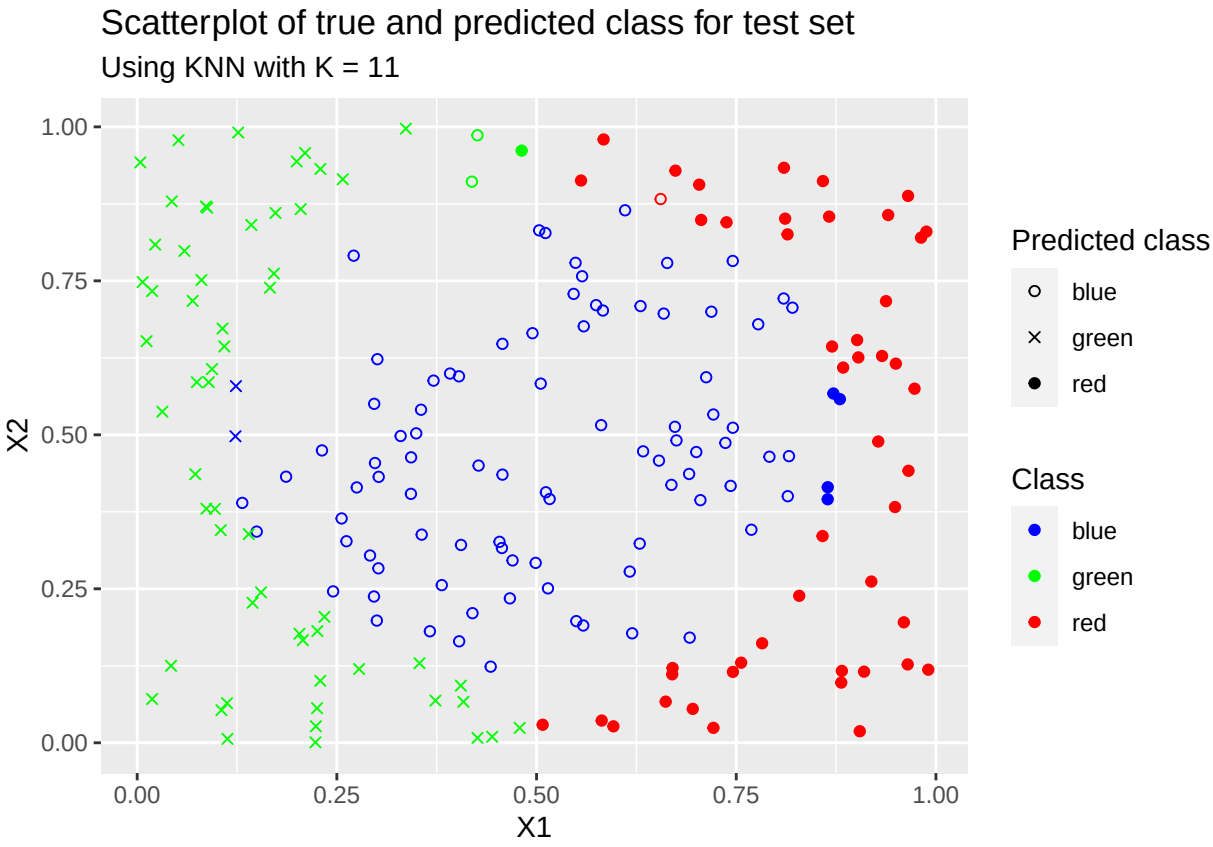
(d) For the value of k that resulted in the smallest test error in part (c) above, make a plot displaying the test observations as well as their true and predicted class labels. Make sure that all axes and points are clearly labeled.

```
min.k <- which(error.results$test.error == min(error.results$test.error))
print(min.k)
```

```
## [1] 11
```

```
predictions <- as.factor(results[min.k, -c(1:3)])
levels(predictions) <- levels(y.test)
best.test.result <- data.frame(x.test, y.test, predictions)

shape.scale.gg <- scale_shape_manual("Predicted class",
```

```
                                    values = c("blue"=1, "green"= 4, "red"=19))

ggplot(data = best.test.result, aes(x = X1, y = X2, color = y.test,
                                    shape = predictions)) +
  geom_point() +
  color.scale.gg +
  shape.scale.gg +
  ggtitle("Scatterplot of true and predicted class for test set",
          subtitle = "Using KNN with K = 11")
```



(e) In this example, what is the Bayes error rate? Justify your answer, and explain how it relates to your findings in (c) and (d).

In this example, the Bayes error rate would be 0. This is because, the Bayes classifier, which is the ideal classifier would be the equations that we used to assign the classes to our training and test sets. We know that $P(Y = red|X) = 1$ if $(X_1 - 0.5)^2 + (X_2 - 0.5)^2 > 0.15$ and $X_1 > 0.5$, $P(Y = green|X) = 1$ if $(X_1 - 0.5)^2 + (X_2 - 0.5)^2 > 0.15$ and $X_1 \leq 0.5$ and $P(Y = blue|X) = 1$ if $(X_1 - 0.5)^2 + (X_2 - 0.5)^2 \leq 0.15$. Using this information, we can always classify all our observations perfectly.

In (d) we can see that the misclassifications are only made in cases that are close to the boundary of the circle $(X_1 - 0.5)^2 + (X_2 - 0.5)^2 = 0.15$, which is the Bayes decision boundary. From part (c) we see that a smaller value of k, which results in higher model flexibility is able to approximate the non-linear decision boundary. The KNN model is trying to approximate the equation of the circle, and the K value in (d) is the one that comes closest to the true boundary.

7. This exercise involves the Boston housing data set, which is part of the ISLR2 library.

(a) How many rows are in this data set? How many columns? What do the rows and columns represent?

```
library(ISLR2)

cat("Rows = ", dim(Boston)[1])
```

```
## Rows =  506
```

```
cat("\nColumns =", dim(Boston)[2])
```

```
##
## Columns = 13
```

Each row represents a suburb and each column represents a feature of the suburb. A detailed description of the dataset from help documentation:

Boston Data

Description: A data set containing housing values in 506 suburbs of Boston.

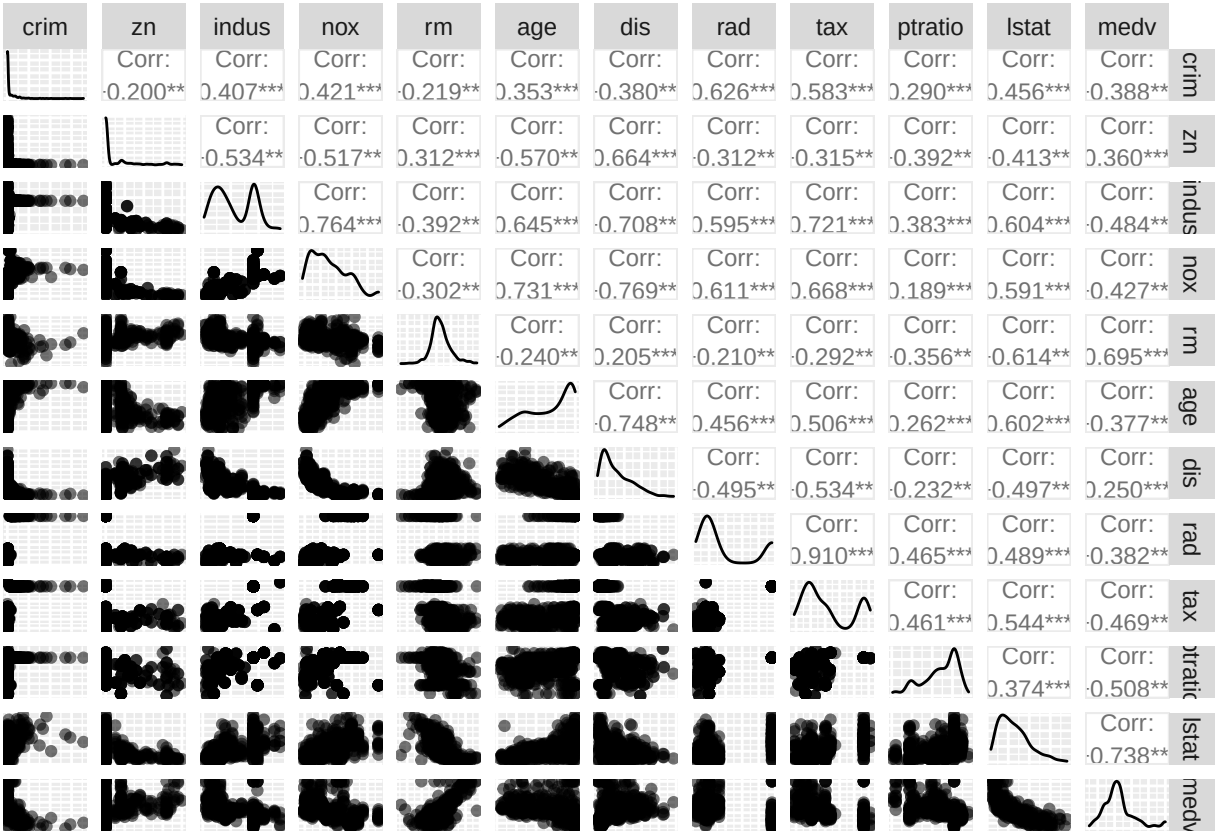
Format: A data frame with 506 rows and 13 variables.

Variables:

- *crim:* per capita crime rate by town.
- *zn:* proportion of residential land zoned for lots over 25,000 sq.ft.
- *indus:* proportion of non-retail business acres per town.
- *chas:* Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
- *nox:* nitrogen oxides concentration (parts per 10 million).
- *rm:* average number of rooms per dwelling.
- *age:* proportion of owner-occupied units built prior to 1940.
- *dis:* weighted mean of distances to five Boston employment centers.
- *rad:* index of accessibility to radial highways.
- *tax:* full-value property-tax rate per $10,000.
- *ptratio:* pupil-teacher ratio by town.
- *lstat:* lower status of the population (percent).
- *medv:* median value of owner-occupied homes in $1000s.

(b) Make some pairwise scatterplots of the predictors (columns) in this data set. Describe your findings.

```
library(GGally)

ggpairs(data = Boston[,-c(4)], aes(alpha = 0.01),
        upper = list(continuous = wrap("cor", size = 3))) +
  theme(axis.line=element_blank(),
        axis.text=element_blank(),
        axis.ticks=element_blank())
```



The categorical variable chas has been omitted form this plot as it does not provide useful information. We make a few observations on apparent linear relationships as they are easiest to discern from the plot:

- Highly positive linear relation (based on correlation coefficient and scatter) between
  - number of rooms per dwelling (rm) and median value of owner occupied homes (medv)
  - lots zoned for 25K sq ft (zn) and weighted distance to 5 Boston employment centers (dis)
  - nitrogen oxide concentration (nox) and proportion of owner occupied units built before 1940 (age)
  - full value property tax per $10K (tax) and proportion of non-retail establishments (indus)
  - crime rate (crim) and accessibility to radial highways (rad)
- Highly negative linear relation (based on correlation coefficient and scatter) between
  - weighted distance to 5 Boston employment centers (dis) and proportion of non-retail establishments (indus)
  - weighted distance to 5 Boston employment centers (dis) and nitrogen oxide concentration (nox)
  - lower status of the population (percent) (lstat) and number of rooms per dwelling (rm)
  - weighted distance to 5 Boston employment centers (dis) and proportion of owner occupied units built before 1940 (age)
  - (lstat) lower status of the population (percent) and median value of owner occupied homes (medv)

(c) Are any of the predictors associated with per capita crime rate? If so, explain the relationship.

Yes, per capita crime rate is related to other variables in the data as follows:

- Increases with decrease in lots zoned for 25K sq ft(zn), weighted distance to 5 Boston employment centers (dis), median value of owner occupied homes (medv)
- Increases with increase in nitrogen oxide concentration (nox), proportion of homes built before 1940 (age), property tax rate (tax), pupil-teacher ratio (ptratio), lower status of the population (percent) (lstat)
- Increases with decrease in rooms per dwelling (rm) up to a certain point and then starts increasing with some outlier observations

(d) Do any of the suburbs of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each predictor.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#Summary statistics for top 10 values of crime rate:
Boston |> top_n(10, crim) |> arrange(desc(crim)) |> select(crim) |>
  summarise(min(crim), mean(crim), max(crim))
```

```
##   min(crim) mean(crim) max(crim)
## 1   25.9406   49.94523   88.9762
```

```
#Summary statistics for top 10 values of tax rate:
Boston |> top_n(10, tax) |> arrange(desc(tax)) |> select(tax) |>
  summarise(min(tax), mean(tax), max(tax))
```

```
##   min(tax) mean(tax) max(tax)
## 1      666  667.6423      711
```

```
#Summary statistics for top 10 values of pupil-teacher ratio:
Boston |> top_n(10, ptratio) |> arrange(desc(ptratio)) |> select(ptratio) |>
  summarise(min(ptratio), mean(ptratio), max(ptratio))
```

```
##   min(ptratio) mean(ptratio) max(ptratio)
## 1         21.2      21.29412           22
```

```
#Summary statistics for crime rate, tax rate and pupil-teacher ratio (entire dataset)
summary(Boston[, c(1,10,11)])
```

```
##       crim              tax           ptratio
##  Min.   : 0.00632   Min.   :187.0   Min.   :12.60
##  1st Qu.: 0.08205   1st Qu.:279.0   1st Qu.:17.40
##  Median : 0.25651   Median :330.0   Median :19.05
##  Mean   : 3.61352   Mean   :408.2   Mean   :18.46
##  3rd Qu.: 3.67708   3rd Qu.:666.0   3rd Qu.:20.20
##  Max.   :88.97620   Max.   :711.0   Max.   :22.00
```

We see that the top 10 values for crime rate are much higher than the 75th percentile value. For tax rate, the 10th highest value is the same as the 75th percentile value (data must have a large number of values with 666) and the pupil-teacher ratio is slightly higher than the 75th percentile value.

The range of crime rate is extremely high, with some neighborhoods having less than < 1 crime per capita while others have almost 89 crimes per capita. There is also a lot of contrast in tax rate where some neighborhoods are taxed almost four times as much as the lowest. While the pupil-teacher ratio has a lower numerical range, a load of 10 extra pupils per teacher makes a huge difference to the individual attention a teacher can provide to a pupil.

(e) How many of the suburbs in this data set bound the Charles river?

```
cat("Number of suburbs bounding the Charles river = ", sum(Boston$chas))
```

```
## Number of suburbs bounding the Charles river =  35
```

(f) What are the mean and standard deviation of the pupil-teacher ratio among the towns in this data set?

```
cat("Mean of ptratio = ", mean(Boston$ptratio))
```

```
## Mean of ptratio =  18.45553
```

```
cat("SD of ptratio = ", sd(Boston$ptratio))
```

```
## SD of ptratio =  2.164946
```

(g) Which suburb of Boston has highest median value of owner-occupied homes? What are the values of the other predictors for that suburb, and how do those values compare to the overall ranges for those predictors? Comment on your findings.

```
#Data for towns with highest median value of owner occupied homes
knitr::kable(Boston |> filter(medv == max(medv)))
```

crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	lstat	medv
1.46336	0	19.58	0	0.6050	7.489	90.8	1.9709	5	403	14.7	1.73	50
1.83377	0	19.58	1	0.6050	7.802	98.2	2.0407	5	403	14.7	1.92	50
1.51902	0	19.58	1	0.6050	8.375	93.9	2.1620	5	403	14.7	3.32	50
2.01019	0	19.58	0	0.6050	7.929	96.2	2.0459	5	403	14.7	3.70	50
0.05602	0	2.46	0	0.4880	7.831	53.6	3.1992	3	193	17.8	4.45	50
0.01381	80	0.46	0	0.4220	7.875	32.0	5.6484	4	255	14.4	2.97	50
0.02009	95	2.68	0	0.4161	8.034	31.9	5.1180	4	224	14.7	2.88	50
0.52693	0	6.20	0	0.5040	8.725	83.0	2.8944	8	307	17.4	4.63	50
0.61154	20	3.97	0	0.6470	8.704	86.9	1.8010	5	264	13.0	5.12	50
0.57834	20	3.97	0	0.5750	8.297	67.0	2.4216	5	264	13.0	7.44	50
0.01501	90	1.21	1	0.4010	7.923	24.8	5.8850	1	198	13.6	3.16	50
4.89822	0	18.10	0	0.6310	4.970	100.0	1.3325	24	666	20.2	3.26	50
5.66998	0	18.10	1	0.6310	6.683	96.8	1.3567	24	666	20.2	3.73	50
6.53876	0	18.10	1	0.6310	7.016	97.5	1.2024	24	666	20.2	2.96	50
9.23230	0	18.10	0	0.6310	6.216	100.0	1.1691	24	666	20.2	9.53	50
8.26725	0	18.10	1	0.6680	5.875	89.6	1.1296	24	666	20.2	8.88	50

There are 16 suburbs with the highest median value of owner occupied homes.

```
#Summary for towns with highest median value of owner occupied homes
summary(Boston |> filter(medv == max(medv)))
```

```
##       crim                zn             indus            chas        
##  Min.   :0.01381   Min.   : 0.00   Min.   : 0.460   Min.   :0.000  
##  1st Qu.:0.40920   1st Qu.: 0.00   1st Qu.: 3.647   1st Qu.:0.000  
##  Median :1.49119   Median : 0.00   Median :18.100   Median :0.000  
##  Mean   :2.70341   Mean   :19.06   Mean   :11.861   Mean   :0.375  
##  3rd Qu.:5.09116   3rd Qu.:20.00   3rd Qu.:18.470   3rd Qu.:1.000  
```

```
##  Max.   :9.23230   Max.   :95.00   Max.   :19.580   Max.   :1.000
##       nox               rm             age              dis
##  Min.   :0.4010   Min.   :4.970   Min.   : 24.80   Min.   :1.130
##  1st Qu.:0.5000   1st Qu.:6.933   1st Qu.: 63.65   1st Qu.:1.351
##  Median :0.6050   Median :7.853   Median : 90.20   Median :2.043
##  Mean   :0.5666   Mean   :7.484   Mean   : 77.64   Mean   :2.586
##  3rd Qu.:0.6310   3rd Qu.:8.100   3rd Qu.: 96.97   3rd Qu.:2.971
##  Max.   :0.6680   Max.   :8.725   Max.   :100.00   Max.   :5.885
##       rad              tax           ptratio          lstat            medv
##  Min.   : 1.00   Min.   :193.0   Min.   :13.00   Min.   :1.730   Min.   :50
##  1st Qu.: 4.75   1st Qu.:261.8   1st Qu.:14.62   1st Qu.:2.967   1st Qu.:50
##  Median : 5.00   Median :403.0   Median :14.70   Median :3.510   Median :50
##  Mean   :10.62   Mean   :415.4   Mean   :16.48   Mean   :4.355   Mean   :50
##  3rd Qu.:24.00   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:4.753   3rd Qu.:50
##  Max.   :24.00   Max.   :666.0   Max.   :20.20   Max.   :9.530   Max.   :50
```

```
#Summary for entire dataset
summary(Boston)
```

```
##       crim                zn              indus            chas
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000
##       nox               rm              age              dis
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       rad              tax           ptratio          lstat
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   : 1.73
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.: 6.95
##  Median : 5.000   Median :330.0   Median :19.05   Median :11.36
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :12.65
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:16.95
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :37.97
##       medv
##  Min.   : 5.00
##  1st Qu.:17.02
##  Median :21.20
##  Mean   :22.53
##  3rd Qu.:25.00
##  Max.   :50.00
```

Following is a summary of how the values of other variables compare to the overall ranges for the towns with the highest median value of owner-occupied homes:

- These towns don't have the lowest or highest crime rate. The mean value is lower but the median is higher. It seems like they experience somewhat high crime.
- More proportion of residential land zoned for lots over 25K sq.ft on an average but the maximum is not in this subset of towns
- 37.5% of these towns bound the Charles river

- The mean rooms per dwelling are higher than the overall mean. The maximum is not a part of this subset
- More owner occupied homes are built before 1940 in this subset, half these towns have around 90.2% homes built before 1940
- The weighted distance to 5 Boston employment centers is lower both in terms of mean value and range for this subset
- The tax rates for this subset are higher on average but not the highest in the data
- The pupil-teacher ratio is lower on average for this subset but not the lowest in the data
- The lower status of the population (percent) is substantially lower for this subset of towns

(h) In this data set, how many of the suburbs average more than six rooms per dwelling? More than eight rooms per dwelling? Comment on the suburbs that average more than eight rooms per dwelling.

```
cat("Suburbs with more than 6 rooms per dwelling on average = ",
    length(which(Boston$rm > 6)))
```

```
## Suburbs with more than 6 rooms per dwelling on average =  333
```

```
cat("Suburbs with more than 8 rooms per dwelling on average = ",
    length(which(Boston$rm > 8)))
```

```
## Suburbs with more than 8 rooms per dwelling on average =  13
```

```
summary(Boston |> filter(rm > 8))
```

```
##       crim              zn           indus            chas        
##  Min.   :0.02009   Min.   : 0.00   Min.   : 2.680   Min.   :0.0000  
##  1st Qu.:0.33147   1st Qu.: 0.00   1st Qu.: 3.970   1st Qu.:0.0000  
##  Median :0.52014   Median : 0.00   Median : 6.200   Median :0.0000  
##  Mean   :0.71879   Mean   :13.62   Mean   : 7.078   Mean   :0.1538  
##  3rd Qu.:0.57834   3rd Qu.:20.00   3rd Qu.: 6.200   3rd Qu.:0.0000  
##  Max.   :3.47428   Max.   :95.00   Max.   :19.580   Max.   :1.0000  
##       nox               rm             age              dis        
##  Min.   :0.4161   Min.   :8.034   Min.   : 8.40   Min.   :1.801  
##  1st Qu.:0.5040   1st Qu.:8.247   1st Qu.:70.40   1st Qu.:2.288  
##  Median :0.5070   Median :8.297   Median :78.30   Median :2.894  
##  Mean   :0.5392   Mean   :8.349   Mean   :71.54   Mean   :3.430  
##  3rd Qu.:0.6050   3rd Qu.:8.398   3rd Qu.:86.50   3rd Qu.:3.652  
##  Max.   :0.7180   Max.   :8.780   Max.   :93.90   Max.   :8.907  
##       rad              tax           ptratio          lstat           medv      
##  Min.   : 2.000   Min.   :224.0   Min.   :13.00   Min.   :2.47   Min.   :21.9  
##  1st Qu.: 5.000   1st Qu.:264.0   1st Qu.:14.70   1st Qu.:3.32   1st Qu.:41.7  
##  Median : 7.000   Median :307.0   Median :17.40   Median :4.14   Median :48.3  
##  Mean   : 7.462   Mean   :325.1   Mean   :16.36   Mean   :4.31   Mean   :44.2  
##  3rd Qu.: 8.000   3rd Qu.:307.0   3rd Qu.:17.40   3rd Qu.:5.12   3rd Qu.:50.0  
##  Max.   :24.000   Max.   :666.0   Max.   :20.20   Max.   :7.44   Max.   :50.0
```

These suburbs have some of the lowest crime rates, pupil-teacher ratios, proportion of non-retail businesses and lower status of the population (percent) and highest median value of owner occupied homes.