LAB0

Mountain Dewds

Individual Assigments

Josh:

This is a photo of me cross country skiing for the first time over winter break. I went with my fiance and her mother in Oregon. It was so much harder than downhill skiing in my opinion!

I want to know what percent of a musician's success is from time practicing, connections with people in industry, and purely luck. I'm not how someone could measure success though, maybe through album sales and net worth?

I would be excited to finally have a full time job and be in industry. Right now I want to be involved with modeling or risk assessment. Ideally I would be at a consulting firm.

My greatest career accomplishment would be running my own statistical consulting firm. I want to be my own boss, but stay working in industry.

I am hoping to learn how to apply my knowledge to real life problems. I hope to use QQQ and learn how to best interpret statistics to give advice for future development. I also want to learn the most effective ways to work within a team.

I have been playing guitar for about 2 years. I used to play the drums in high school, but they took up too much space in my apartment during undergraduate, so I took up guitar. I was in a band in my last year of undergraduate, and I'm hoping to start another soon in the Denver/Boulder area!

Rishi:


16

This is a photo of me playing baseball for the club team here at CU Boulder.

One question I have is how does the pitch sequence in an at-bat(fastball, slider, changeup, curveball) affect the likelihood of a batter getting a hit?

6 months after graduation I would love to be involved in sports so preferably something to do with sports analytics. 5 years down the road I would like to be working for a professional sports organization doing analytics for them to try and help the team win.

My hopes for my greatest career accomplishment is to become a lead analyst for a professional sports team and use those analytics to drive better performance ultimately winning games and potential championships.

I aim to learn more about how to work in a team throughout a full semester and improving my statistical learning skills through collaboration.

Something I am really striving for this semester is to be the best player I can be for the baseball team. There are first team, second team, and third team all americans across the country so I am hoping to be selected to one.

Chase:

Here is an image of me playing golf at a course in Longmont.

One question I would like to answer is how field position, yards to go, and score impact coaches offensive play calls in order to set up a team's defense in the best possible way.

6 months after graduation I would like to be getting my masters in sports analytics and then 5 years after graduation I would like to be using the masters and working as a sports analysis for a professional sports team in Colorado or Florida.

My hope for my career is to obtain a job for the Denver Broncos and help my team become the great team they once were. The idea of working of a sports team sounds like a job where you get excited to go to work everyday.

I am hoping to improve on my teamwork skills while improving my skills in R. I am hoping to learn how to become a vital part of team of data scientists.

I love outdoor activities such as golfing, wake surfing, hunting, and spending time in the mountains.

Will:

This is a photo of me on Gray's peak in the fall this year. My roommate and I climbed the mountain in October, along with Torrey's Peak in the same day. We were expecting it to be cold that high up during the fall, but we weren't expecting all of the wind!

I am very interested in the environment, and where I am from in New Jersey there is an issue with a beetle that is killing a type of hardwood trees. The forest service has started to treat the trees in an effort to keep them from going extinct. I would like to see if their efforts are paying off using statistical methods.

I would think it would be very interested to work for either a statistical consulting firm, or a defense contractor in the future. I would like a job that allows me to apply what I have learned in school to interesting real world topics.

5 years from now I would like to be working my way up the ladder at a good company, or be starting my own business which would be my greatest achievement.

I am hoping to learn a variety of new statistical learning techniques and how to apply them in this class. The more methods that I know, the more valuable I would be to a hiring company in the near future.

Something else interesting about me that is not included in the prompt is that my girlfriend and I are trying to visit all of the national parks in the United States. We went on two road

trips in 2024 and visited 12 of them.

About Our Team

Team Name: Mountain Dewds!

Main Goal for this Semester: We want to continue doing well on the trat exams. We also want to learn how to apply the learning methods in this class.

Applied Portion

Individual

Josh:

Below is the code given in the lab:

```r
library(class)
```

```
Warning: package 'class' was built under R version 4.4.2
```

```r
library(tidyverse)
```

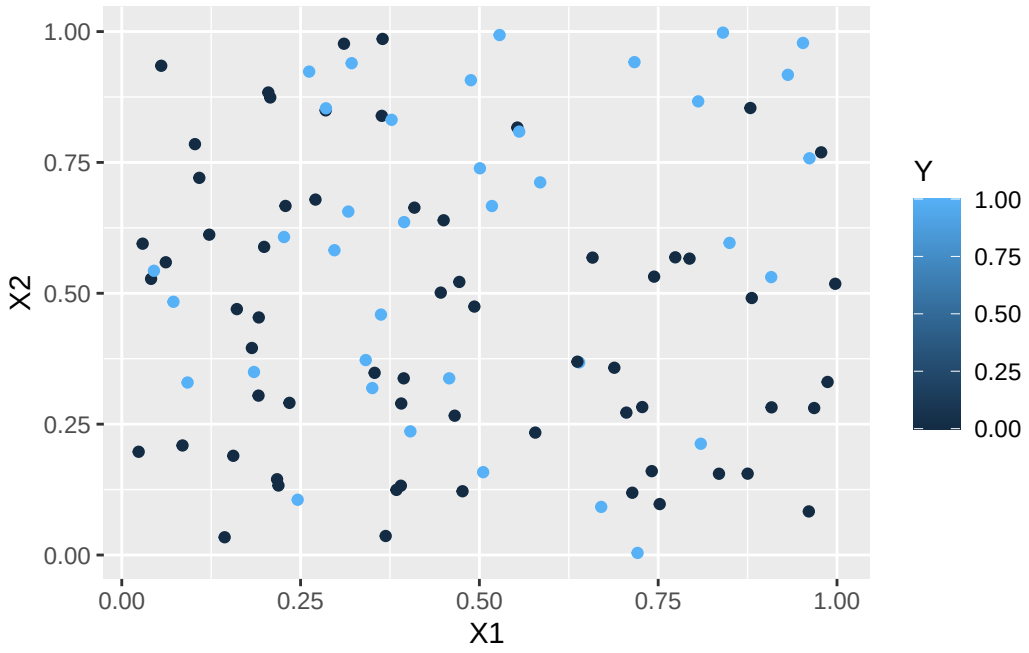
```
Warning: package 'tidyverse' was built under R version 4.4.2

Warning: package 'ggplot2' was built under R version 4.4.2

Warning: package 'tibble' was built under R version 4.4.2

Warning: package 'tidyr' was built under R version 4.4.2

Warning: package 'readr' was built under R version 4.4.2

Warning: package 'purrr' was built under R version 4.4.2

Warning: package 'dplyr' was built under R version 4.4.2

Warning: package 'stringr' was built under R version 4.4.2

Warning: package 'forcats' was built under R version 4.4.2

Warning: package 'lubridate' was built under R version 4.4.2

-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

```r
#Generative model
set.seed(127) #setting a random seed so that we can reproduce everything exactly if we want t
generate_y <- function(x1,x2) { #two input parameters to generate the output y
logit <- x1 -2*x2 -2*x1^2 + x2^2 + 3*x1*x2 +4*x1*x2^2 -3*x1^2*x2
p <- exp(logit)/(1+exp(logit)) #apply the inverse logit function
y <- rbinom(1,1,p) #y becomes a 0 (with prob 1-p) or a 1 with probability p
}
```

```r
# Generate a dataset with 100 points
set.seed(127)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
#I'm going to use a for loop to generate 100 y's
Y <- rep(0,n) #initializing my Y to be a vector of 0's
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y) #How many 0's and 1's were predicted? In this training set, 37% were 1's. However, bet
```

```
[1] 37
```

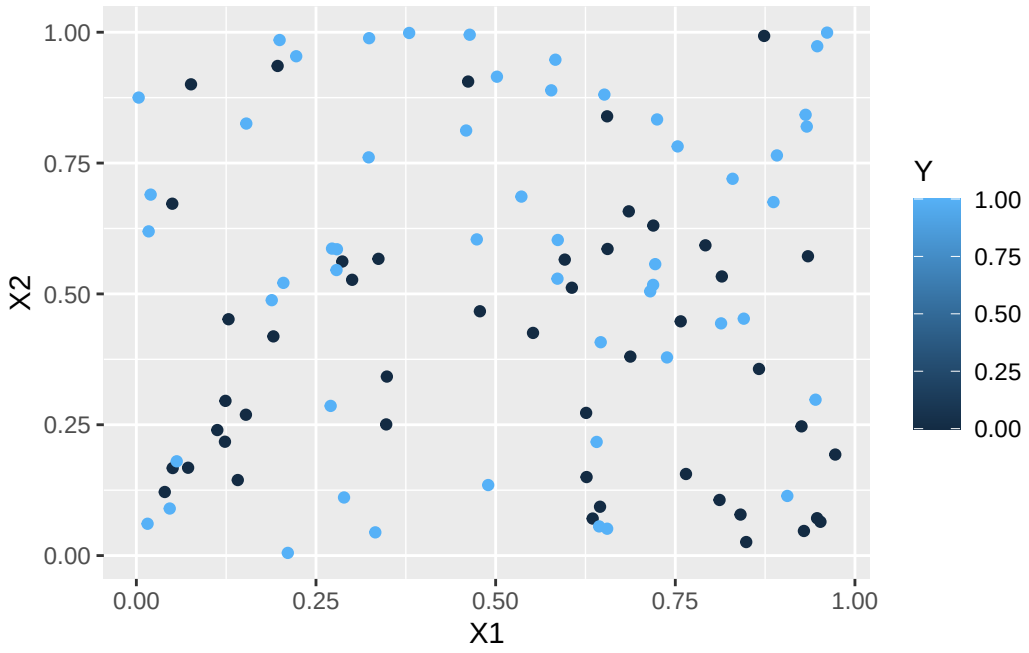
```r
training <- cbind(X1,X2,Y) #combining all of my variables into a training dataset
ggplot(data=training, aes(x=X1, y=X2, color=Y)) +
geom_point()
```




```
# Create the training dataset as above using seed=127
# Create a testing dataset using seed=128
set.seed(128)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
#I'm going to use a for loop to generate 100 y's
Y <- rep(0,n) #initializing my Y to be a vector of 0's
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y) #53 1's, which is much closer to the 51.5% true rate
```

```
[1] 53
```

```
testing <- cbind(X1,X2,Y)
#Let's plot the test set. Does it look like the training set? Yeah, looks similar.
ggplot(data=testing, aes(x=X1, y=X2, color=Y)) +
geom_point()
```



1. Given the training set (seed=127) and the testing set (seed=128), fit KNN on two different values of k.

First I tried with k=3 and then k=50

```
knn3predictions <- knn(train=training[,1:2], cl=training[,3], test = testing[,1:2], k = 3)

knn3predictions <- data.frame(testing,knn3predictions)


knn50predictions <- knn(train=training[,1:2], cl=training[,3], test = testing[,1:2], k = 50)

knn50predictions<- data.frame(testing,knn50predictions)
```

2. Calculate the misclassification rate for each k. If you don't know how to do this, ask a teammate or the professor.

I tried to do this directly, I'm not sure if there is a function in R that does this. Below are the misclassification rates for k=3 and k =50 respectively.

```
knn3misclassificationrate=sum(knn3predictions[,3]!=knn3predictions[,4])/length(knn3predictio
knn3misclassificationrate
```

```
[1] 0.45
```

```
knn50misclassificationrate=sum(knn50predictions[,3]!=knn50predictions[,4])/length(knn50predi
knn50misclassificationrate
```

```
[1] 0.53
```

3. If possible, plot the decision boundaries for your k values.

I had trouble doing this, so I did not present it here.

4. Summarize the Q1, Q2, and Q3 aspects of this "project." Use your imagination. Everyone should have a different scenario.

A company wants to launch a new hot dog campaign over blu-ray ads in blue-rays and products in the top shelves of stores. The slogan is "hot dogs are the best sandwich!" They collected data on where they should market this, including respondents height and amount spent on blu-ray dvds. Q1 in this case could be data collected from a survey of individuals who think hot dogs are a sandwich over their amount spent on blu-ray dvds in the past year and height. The X1 column is each observation's amount spent in dollars on blu-ray dvds and X2 is each respondents height. The class, Y, is whether or not the respondent believes a hot dog is a sandwich. The Q2 aspect of this project is using knn to identify decision boundaries on height and amount spent on blu-rays to find any relationship. In Q3, we would conclude that there is likely no relationship between these variables and the class, so this company should not invest differently in blu-ray dvd ads or markets that prioritize tall individuals.

5. Think about and discuss with your team some of the bonus questions below.

Rishi:

Questions 1 and 2 down below:

Step 1: Loading the libraries

```
library(class)
library(tidyverse)
```

Step 2: Generating the training data

```r
set.seed(127)
n <- 100
X1 <- runif(n, 0, 1)
X2 <- runif(n, 0, 1)

generate_y <- function(x1, x2) {
  logit <- x1 - 2 * x2 - 2 * x1^2 + x2^2 + 3 * x1 * x2 + 4 * x1 * x2^2 - 3 * x1^2 * x2
  p <- exp(logit) / (1 + exp(logit))
  y <- rbinom(1, 1, p)
  return(y)
}

Y <- sapply(1:n, function(i) generate_y(X1[i], X2[i]))
training <- data.frame(X1, X2, Y)
```

Step 3: Generating the testing data

```r
set.seed(128)
X1_test <- runif(n, 0, 1)
X2_test <- runif(n, 0, 1)
Y_test <- sapply(1:n, function(i) generate_y(X1_test[i], X2_test[i]))
testing <- data.frame(X1 = X1_test, X2 = X2_test, Y = Y_test)
```

Step 4: Picking k = 4 and k =8

```r
k1 <- 4
k2 <- 8

pred_k1 <- knn(train = training[, c("X1", "X2")],
               test = testing[, c("X1", "X2")],
               cl = training$Y,
               k = k1)

pred_k2 <- knn(train = training[, c("X1", "X2")],
               test = testing[, c("X1", "X2")],
               cl = training$Y,
               k = k2)
```

Step 5: Calculating the Classification Rates

```r
misclass_k1 <- mean(pred_k1 != testing$Y)
misclass_k2 <- mean(pred_k2 != testing$Y)

cat("Misclassification rate for k =", k1, "is", misclass_k1, "\n")
```

```
Misclassification rate for k = 4 is 0.41
```

```r
cat("Misclassification rate for k =", k2, "is", misclass_k2, "\n")
```

```
Misclassification rate for k = 8 is 0.45
```

Question 3: I didn't plot the decision boundary

Question 4:

Q1: Y represents whether a home run is hit, X1 is the pitch speed, and X2 is the launch angle of the bat. This is important because it can provide insight as to how to improve batting approaches and understand how pitchers decide what pitches to throw to try and get the batter out.

Q2: I used KNN for k = 4 and k = 8 to predict home run outcomes for a test data set of pitch speeds and launch angles.

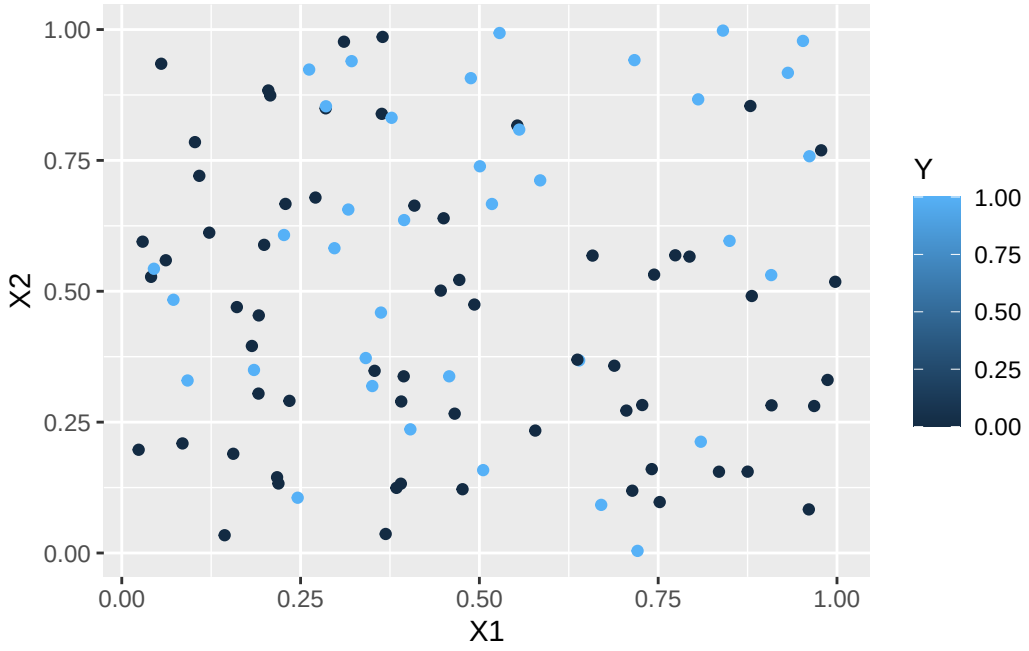
Q3: For X1= 0.5 and X2 = 0.5, the model with k = 4 predicted a home run and the model with k = 8 didn't. This shows how different k values affect the decision boundary.

Chase:

```r
set.seed(127) #setting a random seed so that we can reproduce everything exactly if we want t
generate_y <- function(x1,x2) { #two input parameters to generate the output y
logit <- x1 -2*x2 -2*x1^2 + x2^2 + 3*x1*x2 +4*x1*x2^2 -3*x1^2*x2
p <- exp(logit)/(1+exp(logit)) #apply the inverse logit function
y <- rbinom(1,1,p) #y becomes a 0 (with prob 1-p) or a 1 with probability p
}
set.seed(127)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
#I'm going to use a for loop to generate 100 y's
Y <- rep(0,n) #initializing my Y to be a vector of 0's
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y) #How many 0's and 1's were predicted? In this training set, 37% were 1's. However, be
```

```
[1] 37
```

```
training <- cbind(X1,X2,Y) #combining all of my variables into a training dataset
ggplot(data=training, aes(x=X1, y=X2, color=Y)) +
geom_point()
```

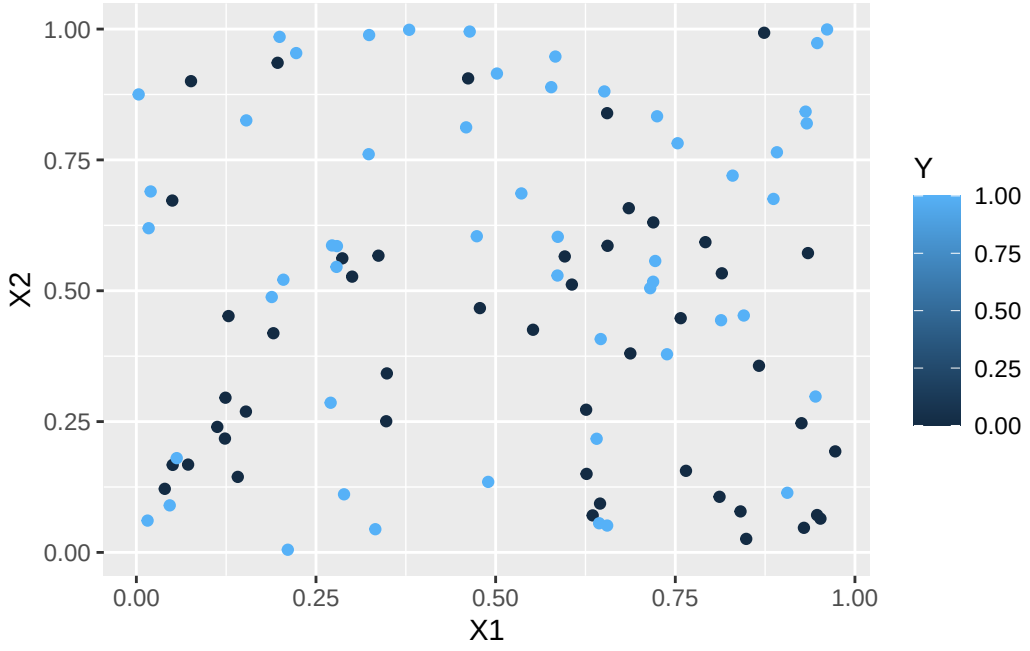



```
set.seed(128)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
#I'm going to use a for loop to generate 100 y's
Y <- rep(0,n) #initializing my Y to be a vector of 0's
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y) #53 1's, which is much closer to the 51.5% true rate
```

```
[1] 53
```

```
testing <- cbind(X1,X2,Y)
#Let's plot the test set. Does it look like the training set? Yeah, looks similar.
```

```
ggplot(data=testing, aes(x=X1, y=X2, color=Y)) +
geom_point()
```



1. I decided to compare k=1 and k=5

```
k = 1
knn1_predictions <- knn(train = training[, 1:2], cl = training[, 3], test = testing[, 1:
knn1 <- data.frame(testing, Predicted = knn1_predictions)

k = 5
knn5_predictions <- knn(train = training[, 1:2], cl = training[, 3], test = testing[, 1:
knn5 <- data.frame(testing, Predicted = knn5_predictions)
```

2.
```
misclass_rate_k1 <- mean(knn1$Predicted != knn1$Y)
misclass_rate_k5 <- mean(knn5$Predicted != knn5$Y)

cat("Misclassification Rate for k=1:", misclass_rate_k1, "\n")
```

```
Misclassification Rate for k=1: 0.44
```

```
cat("Misclassification Rate for k=5:", misclass_rate_k5, "\n")
```

```
Misclassification Rate for k=5: 0.38
```

3. I was unable to figure out how to do this.

4. Q1: A sports team wants to decide if they should trade a player based on their performance and salary. They collected data on factors such as the player's average points per game (PPG) and their annual salary. Q1 in this case X1 being the player's PPG and X2 being their salary. The class, Y, is whether or not the player should be traded. This is important because it helps the team determine if a player's performance justifies their salary and whether a trade would be beneficial for the team's overall strategy.

Q2: The model uses the collected data to predict whether a player should be traded based on their PPG and salary, looking for patterns that help identify which players are likely to be traded.

Q3: For a player with X1 = 15 (PPG) and X2 = 10 million (salary), the model predicted that they should not be traded. This shows that performance and salary alone may not always result in a trade, and other factors might be influencing the decision.

Will:

Generate the Data:

```
set.seed(127)
generate_y <- function(x1,x2) {
logit <- x1 -2*x2 -2*x1^2 + x2^2 + 3*x1*x2 +4*x1*x2^2 -3*x1^2*x2
p <- exp(logit)/(1+exp(logit))
y <- rbinom(1,1,p)
}
```

```
set.seed(127)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
Y <- rep(0,n)
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y)
```

```
[1] 37
```

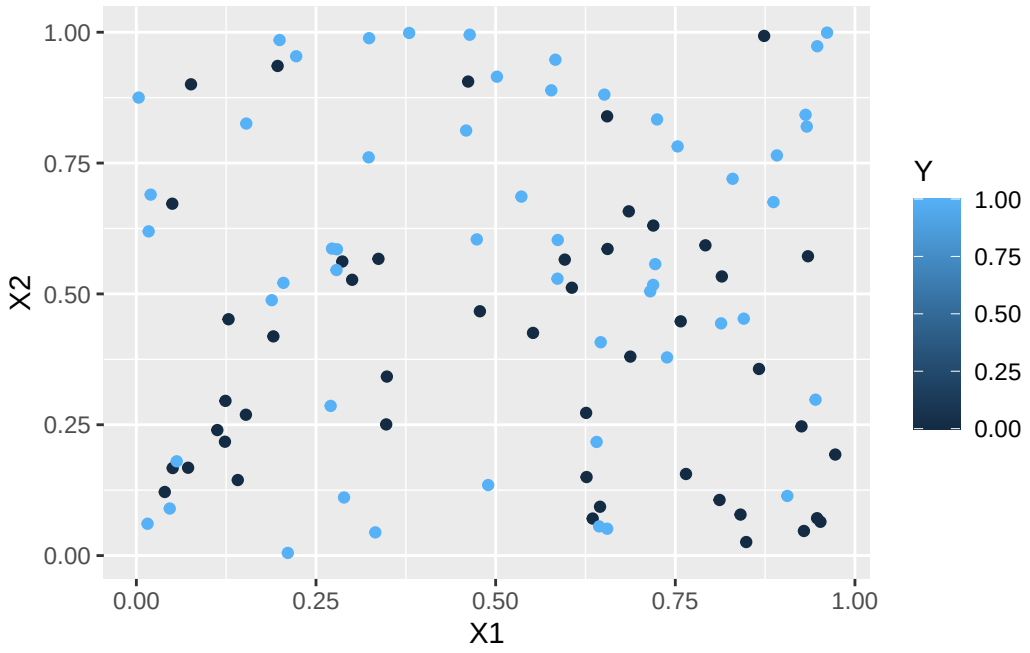
```
training <- cbind(X1, X2,Y)
training <- as.data.frame(training)
```

```
set.seed(128)
generate_y <- function(x1,x2) {
logit <- x1 -2*x2 -2*x1^2 + x2^2 + 3*x1*x2 +4*x1*x2^2 -3*x1^2*x2
p <- exp(logit)/(1+exp(logit))
y <- rbinom(1,1,p)
}
```

```
set.seed(128)
n = 100
X1 <- runif(n,0,1)
X2 <- runif(n,0,1)
Y <- rep(0,n)
for (i in 1:n) {
Y[i] <- generate_y(X1[i],X2[i])
}
sum(Y)
```

```
[1] 53
```

```
testing <- cbind(X1,X2,Y)
testing <- as.data.frame(testing)
ggplot(data = testing, aes(x=X1, y=X2, color = Y)) +
  geom_point()
```



1.

```
testing$K2_pred <- as.data.frame(knn(train = training[1:2], test = testing[1:2], cl = traini
testing$K6_pred <- as.data.frame(knn(train = training[1:2], test = testing[1:2], cl = traini
```

2.

```
misclassification_K2 <- mean(testing$K2_pred != testing$Y)
print(misclassification_K2)
```

```
[1] 0.42
```

```
misclassification_K6 <- mean(testing$K6_pred != testing$Y)
print(misclassification_K6)
```

```
[1] 0.35
```

The misclassification for K = 2 and K = 6 was 0.41 for both. This means these models are equally as bad and I don't recommend using either one

3.

I had a lot of trouble trying to create the decision boundaries for these.

4.

Q1: I will use the example I gave in my individual introduction in this lab. I want to test if a tree will live or die due to the the treatment it received against beetles. Y will equal whether the tree remains healthy or dies based on the tree's age (X1) and the tree's size (X2). We should care because the trees with the best chances of surviving should get the most possible resources so the species does not go extinct.

Q2:

```
library(class)
new_point <- data.frame(X1 = 0.5, X2 = 0.5)
prediction_k2 <- knn(
  train = training[1:2],
  test = new_point,
  cl = training$Y,
  k = 2
)
prediction_k6 <- knn(
  train = training[1:2],
  test = new_point,
  cl = training$Y,
  k = 6
)

print(paste("Prediction for k = 2:", prediction_k2))
```

```
[1] "Prediction for k = 2: 0"
```

```
print(paste("Prediction for k = 6:", prediction_k6))
```

```
[1] "Prediction for k = 6: 0"
```

This prediction shows that for both K=2 and K=6, the model predicts that the tree will die if its age and size are both 0.5.

Q3: This does have some ethical concerns as only some areas of forest may be saved because of optimal tree size, while other forests might die. This could impact animal habitat and could lead to a certain type of animal dying as well.

Team

Which K value is best from what we tested?

After comparing our individual portions, we decided that a k value of 5 was best as it had a misclassification rate of 0.38, which was the lowest of the k values we tested. We saw that k values after k=5 began to have a larger misclassification rate, so this further demonstrated that a k of 5 seemed to have the best misclassifcation rate.