1. Consider the toy dataset below which shows if 4 subjects have diabetes or not, along with two diagnostic measurements.

Preg	**BP**	**HasDiabetes**	**Preg.Norm**	**BP.Norm**
2	74	No	0.5	1.0
3	58	Yes	1.0	0.2
2	58	Yes	0.5	0.2
1	54	No	0.0	0.0
p2	70	?	0.5	0.8

a. Which variable is the “Class” variable?

The Class variable is HasDiabetes

b. Normalize the Preg and BP values by scaling the minimum-maximum range of each column to 0-1. Fill in the empty columns in the table.

c. Predict whether a subject with Preg=2, BP=70 will have diabetes using the 1-NN algorithm and

i. Using Euclidean distance on the original variables

Nearest distance: 0

Nearest neighbor: Row 5

ii. Using Euclidean distance on the normalized variables

Nearest distance: 69.0163

Nearest neighbor: Row 1

For each of these cases, give the nearest distance, nearest neighbor (e.g., “Row 1” or “Row 2”), and prediction.

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

first <- c(2,3,2,1,2)
second <- c(74,58,58,54,70)

df <- data.frame(first, second)

normalized<-function(y) {

  x<-y[!is.na(y)]

  x<-(x - min(x)) / (max(x) - min(x))

  y[!is.na(y)]<-x

  return(y)
}

apply(df[,c(1,2)],2,normalized)

a0 <- c(2, 70)
a1 <- c(2, 74)
a2 <- c(3, 58)
a3 <- c(2, 58)
a4 <- c(1, 54)
a5 <- c(2, 70)

d1 <- sqrt(sum((a0-a1)^2))
d1
d2 <- sqrt(sum((a0-a2)^2))
d2
d3 <- sqrt(sum((a0-a3)^2))
d3
d4 <- sqrt(sum((a0-a4)^2))
d4
d5 <- sqrt(sum((a0-a5)^2))
d5

a0 <- c(2, 70)
a1 <- c(0.5, 1.0)
a2 <- c(1.0, 0.2)
a3 <- c(0.5, 0.2)
a4 <- c(0.0, 0.0)
a5 <- c(0.5, 0.8)

d1 <- sqrt(sum((a0-a1)^2))
d1
d2 <- sqrt(sum((a0-a2)^2))
d2
d3 <- sqrt(sum((a0-a3)^2))
d3
d4 <- sqrt(sum((a0-a4)^2))
d4
d5 <- sqrt(sum((a0-a5)^2))
d5

```

```r
pima <- read_csv("pima-indians-diabetes-resampled.csv")
pima



pima <- filter(pima, Glucose > 0)

normalize <- function(x) { return ((x-min(x)) / (max(x)-min(x)) )}
pimaNorm <- pima %>%
mutate(Preg.norm=normalize(Preg),Pedigree.norm=normalize(Pedigree),Glucose.norm=normalize(Glucos


trainIndex <- sample(1:500)

trainfeatures <- pimaNorm[trainIndex, c(1,2,3,4,5,6,7,8,9,10,11,12)]
trainlabels <- pimaNorm[trainIndex, c(1,2,3,4,5,6,7,8,9,10,11,12)]

testIndex <- setdiff(1:nrow(pimaNorm), trainIndex)

testfeatures <- pimaNorm[testIndex, c(1,2,3,4,5,6,7,8,9,10,11,12)]
testlabels <- pimaNorm[testIndex, c(1,2,3,4,5,6,7,8,9,10,11,12)]

trainfeatures
trainlabels
testfeatures
testlabels

trainfeatures <- pimaNorm[trainIndex, c(10,11,12)]
trainlabels <- pimaNorm[trainIndex, c(10,11,12)]

testfeatures <- pimaNorm[testIndex, c(10,11,12)]
testlabels <- pimaNorm[testIndex, c(10,11,12)]

trainfeatures
trainlabels
testfeatures
testlabels

k1 <- knn(train = trainfeatures, test = testfeatures,
                 cl = trainlabels, k=1)
table(testlabels, k1)

k5 <- knn(train = trainfeatures, test = testfeatures,
          cl = trainlabels, k=5)
table(testlabels, k5)

k11 <- knn(train = trainfeatures, test = testfeatures,
          cl = trainlabels, k=11)
table(testlabels, k11)
```