```
######2

library(ISLR2)
data("Caravan")

#a
train = Caravan[1:1000,]
test = Caravan[-(1:1000),]

#b
library(randomForest)

err = NULL
for(i in seq(6,26, by = 2)){
  tree <- randomForest(Purchase~., Caravan, subset = c(1:1000),
                       mtry = i, importance = TRUE)
  yhatPur <- predict(tree , newdata = test)
  err[i] =  1 - sum(diag(table(test$Purchase, yhatPur)))/
sum(table(test$Purchase, yhatPur))
  print(i)
}
plot(err, xlim = c(6,26))

#Choose 10
tree <- randomForest(Purchase~., Caravan, subset = c(1:1000),
                       mtry = 10, importance = TRUE)
#c
pred <- predict(tree, newdata = test)
head(pred)

#d
table(pred, test$Purchase)[2,2]/sum(table(pred, test$Purchase)[2,])
#.19444

#e
library(class)
standardized.X <- scale(Caravan[, -86])
tester <- 1:1000
train.X <- standardized.X[-tester, ]
test.X <- standardized.X[tester, ]
train.Y <- Caravan$Purchase[-tester]
test.Y <- Caravan$Purchase[tester]
knn.pred <- knn(train.X, test.X, train.Y, k = 10)
table(knn.pred, test.Y)
mean(knn.pred == test.Y) ## For KNN

mean(pred == test$Purchase) #For Random Tree

#The performances of the two are very similar but the KNN edges it out
slightly


#3
library(palmerpenguins)
```

```
library(mice)
#a
imp <- mice(penguins, print = FALSE, m = 10)
data1 = complete(imp)
data2 = complete(imp, 'all')

#b
library(datasets)

pr.out <- prcomp(data1[-c(1,2,7,8)], scale = TRUE)

pr.var <- pr.out$sdev^2
pr.var

pve <- pr.var / sum(pr.var)
pve

par(mfrow = c(1, 2))
plot(pve, xlab = "Principal Component",
     ylab = "Proportion of Variance Explained", ylim = c(0, 1),
     type = "b")
plot(cumsum(pve), xlab = "Principal Component",
     ylab = "Cumulative Proportion of Variance Explained",
     ylim = c(0, 1), type = "b")
dev.off()

biplot(pr.out, scale = 0)

#Choose 2 components

#c
#I chose 2 because by doing so I am only losing about 12% of explained
variability
#It satisfies the elbow rule
#The variance is .77 but that is close enough to 1 given that we are not
losing a lot of explained variability
```