lab2_assignment1

Cui Qingxuan

2024-11-28

1. Data Preparation

```
generateTrainData = function(condition){
  x1 = runif(100)
  x2 = runif(100)
  trdata = cbind(x1,x2)
  y = as.numeric(eval(parse(text = condition)))
  trlabels = as.factor(y)
  return(cbind(trdata, trlabels))
}

generateTestData = function(condition){
  set.seed(1234)
  x1 = runif(1000)
  x2 = runif(1000)
  tedata = cbind(x1,x2)
  y = as.numeric(eval(parse(text = condition)))
  telabels = as.factor(y)
  return(cbind(tedata, telabels))
}
```

2. Pipeline Construction

```
library(randomForest)
```

```
## randomForest 4.7-1.2
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
trainFor3000Times = function(node_size, condition){
  test_data = generateTestData(condition)
  test_data[,3] = as.factor(test_data[,3])
  ntrees = c(1, 10, 100)
  rf_list = list()
  mes = vector()
  for (j in 1:3){
    for(i in 1:1000){
      train_data = generateTrainData(condition)
      rf = randomForest(as.factor(trlabels)~., data = train_data, ntrees = ntrees[j], nodesize = node_s
      output = predict(rf, newdata = test_data)
      me = mean(output != test_data[,3])
      mes = c(me, mes)
```

```
    }
    # report here
    cat("Those 1000 random forests have ", ntrees[j], " trees:\n")
    cat("The mean of misclassification error: ", mean(mes), "\n")
  }
}
```

3. Train

```
cat("Node Size = 25     Condition: x1<x2\n")
```

```
## Node Size = 25     Condition: x1<x2
```

```
trainFor3000Times(node_size = 25, condition = "x1<x2")
```

```
## Those 1000 random forests have  1  trees:
## The mean of misclassification error:  0.109666
## Those 1000 random forests have  10  trees:
## The mean of misclassification error:  0.1093815
## Those 1000 random forests have  100  trees:
## The mean of misclassification error:  0.1092253
```

```
cat("Node Size = 12     Condition: x1<0.5\n")
```

```
## Node Size = 12     Condition: x1<0.5
```

```
trainFor3000Times(node_size = 12, condition = "x1<0.5")
```

```
## Those 1000 random forests have  1  trees:
## The mean of misclassification error:  0.006055
## Those 1000 random forests have  10  trees:
## The mean of misclassification error:  0.006074
## Those 1000 random forests have  100  trees:
## The mean of misclassification error:  0.006091
```

```
cat("Node Size = 12     Condition: (x1<0.5 & x2<0.5)|(x1>0.5 & x2>0.5)\n")
```

```
## Node Size = 12     Condition: (x1<0.5 & x2<0.5)|(x1>0.5 & x2>0.5)
```

```
trainFor3000Times(node_size = 12, condition = "(x1<0.5 & x2<0.5)|(x1>0.5 & x2>0.5)")
```

```
## Those 1000 random forests have  1  trees:
## The mean of misclassification error:  0.073029
## Those 1000 random forests have  10  trees:
## The mean of misclassification error:  0.072608
## Those 1000 random forests have  100  trees:
## The mean of misclassification error:  0.072466
```