Task 1

Read the dataset

Code:

```
setwd('/Users/duwonha/Desktop/University/Second year/T2/Big data/Assignment2')
drugdata <- read.csv('drug200.csv', stringsAsFactors = TRUE)
```

Task 2

Dataset split into training and test sets

Code:

```
training_size <- floor(0.8 * nrow(drugdata))
set.seed(101)
train_int <- sample(seq_len(nrow(drugdata)), size = training_size)
trainingSet <- drugdata[train_int, ]
testSet <- drugdata[-train_int, ]
```

Task 3

Importing library

Code:

```
library(ISLR)
data(package = "ISLR")
require(tree)
```

Tree Construnction

Code:

```
tree_accuracy <- tree(formula = Drug ~ Age+Sex+BP+Cholesterol+Na_to_K, data =
trainingSet)
```

Tree Plot

Code:

```
plot(tree_accuracy)
text(tree_accuracy, pretty = 0)
```

Accuracy Calculation

Code:

```
tree_pred = predict(tree_accuracy, drugdata[-train_int,], type = 'class')
with(drugdata[-train_int,], table(tree_pred, Drug))
```

Task 4

Cross-validation and Tree Construction

Code:

```
drug.cv = cv.tree(tree_accuracy, FUN = prune.misclass)
drug.cv > plot(drug.cv)
drug.cv
```

#Once you run the drug.cv, you can see deviation of size 6 is 1 which means size 6 is the most accurate one. Size 6 will be used for making prune. (best = 6)

```
$size
[1] 6 4 3 2 1

$dev
[1]  1 19 26 43 84
```

prune.drug = prune.misclass(tree_accuracy, best = 6)

Tree plot

Code:

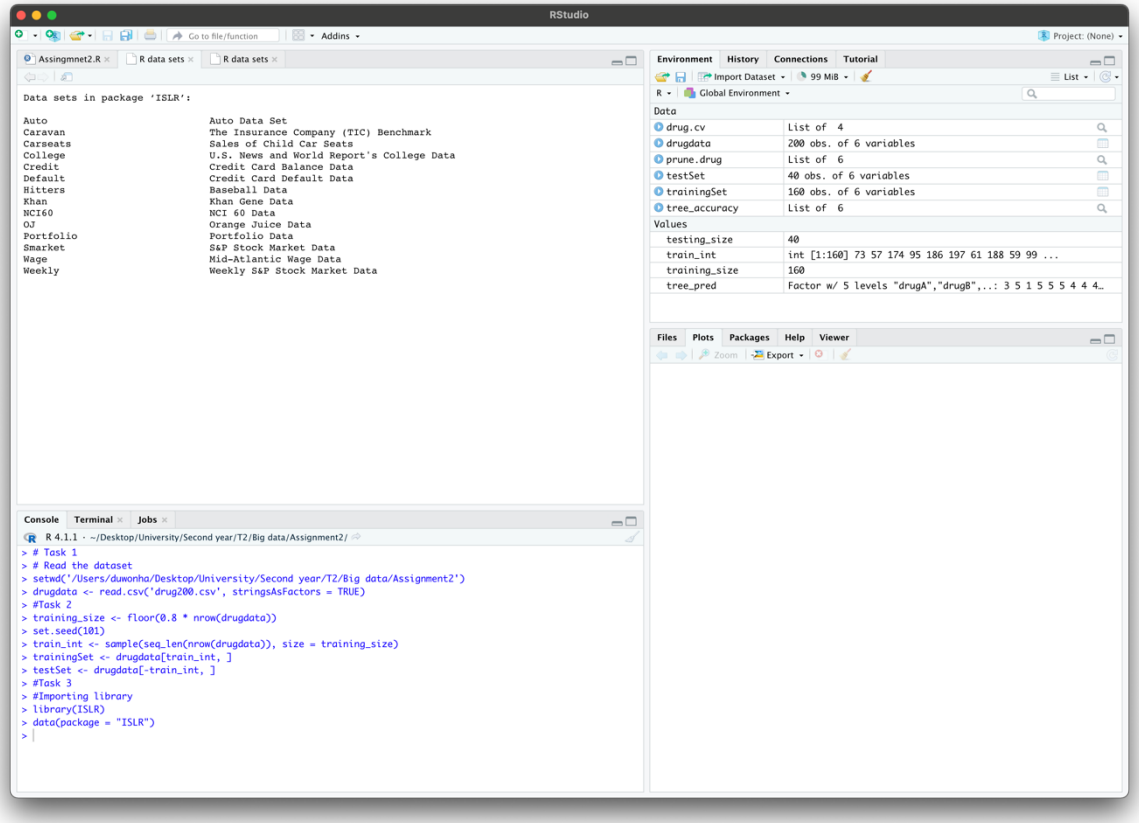
```
plot(prune.drug)
text(prune.drug, pretty=0)
```

Accuracy Calculation

Code:

```
tree_pred = predict(prune.drug, drugdata[-train_int,], type = "class")
with(drugdata[-train_int,], table(tree_pred, Drug))
```

Screenshot



RStudio

Assingmnet2.R | R data sets | R data sets

```
# Task 1
# Read the dataset
setwd('/Users/duwonha/Desktop/University/Second year/T2/Big data/Assignment2')
drugdata <- read.csv('drug200.csv', stringsAsFactors = TRUE)

#Task 2
training_size <- floor(0.8 * nrow(drugdata))
set.seed(101)
train_int <- sample(seq_len(nrow(drugdata)), size = training_size)
trainingSet <- drugdata[train_int, ]
testSet <- drugdata[-train_int, ]
#Task 3
#Importing library
library(ISLR)
data(package = "ISLR")
require(tree)
# Tree Construnction
tree_accuracy <- tree(formula = Drug ~ Age+Sex+BP+Cholesterol+Na_to_K, data = trainingSet)
#Tree Plot
plot(tree_accuracy)
text(tree_accuracy, pretty = 0)
# Accuracy Calculation
tree_pred = predict(tree_accuracy, drugdata[-train_int,], type = 'class')
with(drugdata[-train_int,], table(tree_pred, Drug))

#Task 4
# Cross-validation and Tree Construction
drug.cv = cv.tree(tree_accuracy, FUN = prune.misclass)
drug.cv > plot(drug.cv)
prune.drug = prune.misclass(tree_accuracy, best = length(tree_accuracy))
# Tree plot
```

22:1 (Top Level) R Script

Console | Terminal | Jobs

R 4.1.1 · ~/Desktop/University/Second year/T2/Big data/Assignment2/

```
> #Task 2
> training_size <- floor(0.8 * nrow(drugdata))
> set.seed(101)
> train_int <- sample(seq_len(nrow(drugdata)), size = training_size)
> trainingSet <- drugdata[train_int, ]
> testSet <- drugdata[-train_int, ]
> #Task 3
> #Importing library
> library(ISLR)
> data(package = "ISLR")
> require(tree)
Loading required package: tree
> # Tree Construnction
> tree_accuracy <- tree(formula = Drug ~ Age+Sex+BP+Cholesterol+Na_to_K, data = trainingSet)
> #Tree Plot
> plot(tree_accuracy)
> text(tree_accuracy, pretty = 0)
>
```

Environment | History | Connections | Tutorial

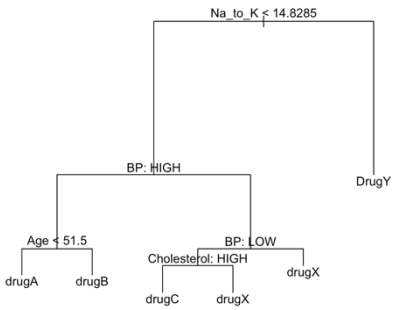
Import Dataset · 131 MiB · List

R · Global Environment

Data	
drug.cv	List of 4
drugdata	200 obs. of 6 variables
prune.drug	List of 6
testSet	40 obs. of 6 variables
trainingSet	160 obs. of 6 variables
tree_accuracy	List of 6
Values	
testing_size	40
train_int	int [1:160] 73 57 174 95 186 197 61 188 59 99 ...
training_size	160
tree_pred	Factor w/ 5 levels "drugA","drugB",..: 3 5 1 5 5 5 4 4 4…

Files | Plots | Packages | Help | Viewer

Zoom · Export · Publish



RStudio

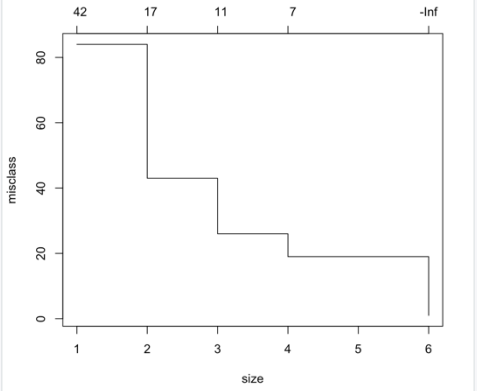
```
drugdata <- read.csv('drug200.csv', stringsAsFactors = TRUE)

#Task 2
training_size <- floor(0.8 * nrow(drugdata))
set.seed(101)
train_int <- sample(seq_len(nrow(drugdata)), size = training_size)
trainingSet <- drugdata[train_int, ]
testSet <- drugdata[-train_int, ]
#Task 3
#Importing library
library(ISLR)
data(package = "ISLR")
require(tree)
# Tree Construnction
tree_accuracy <- tree(formula = Drug ~ Age+Sex+BP+Cholesterol+Na_to_K, data = trainingSet)
#Tree Plot
plot(tree_accuracy)
text(tree_accuracy, pretty = 0)
# Accuracy Calculation
tree_pred = predict(tree_accuracy, drugdata[-train_int,], type = 'class')
with(drugdata[-train_int,], table(tree_pred, Drug))

#Task 4
# Cross-validation and Tree Construction
drug.cv = cv.tree(tree_accuracy, FUN = prune.misclass)
drug.cv > plot(drug.cv)
prune.drug = prune.misclass(tree_accuracy, best = length(tree_accuracy))
# Tree plot
plot(prune.drug)
text(prune.drug, pretty=0)
# Accuracy Calculation
```

```
> text(tree_accuracy, pretty = 0)
> # Accuracy Calculation
> tree_pred = predict(tree_accuracy, drugdata[-train_int,], type = 'class')
> with(drugdata[-train_int,], table(tree_pred, Drug))
          Drug
tree_pred drugA drugB drugC drugX DrugY
    drugA     6     1     0     0     0
    drugB     0     4     0     0     0
    drugC     0     0     2     0     0
    drugX     0     0     0    12     0
    DrugY     0     0     0     0    15
> #Task 4
> # Cross-validation and Tree Construction
> drug.cv = cv.tree(tree_accuracy, FUN = prune.misclass)
> drug.cv > plot(drug.cv)
logical(0)
> prune.drug = prune.misclass(tree_accuracy, best = length(tree_accuracy))
>
```

Environment	
Data	
drug.cv	List of 4
drugdata	200 obs. of 6 variables
prune.drug	List of 6
testSet	40 obs. of 6 variables
trainingSet	160 obs. of 6 variables
tree_accuracy	List of 6
Values	
testing_size	40
train_int	int [1:160] 73 57 174 95 186 197 61 188 59 99 ...
training_size	160
tree_pred	Factor w/ 5 levels "drugA","drugB",..: 3 5 1 5 5 5 4 4 4…



RStudio

Assingmnet2.R* | R data sets | R data sets

```
set.seed(101)
train_int <- sample(seq_len(nrow(drugdata)), size = training_size)
trainingSet <- drugdata[train_int, ]
testSet <- drugdata[-train_int, ]
#Task 3
#Importing library
library(ISLR)
data(package = "ISLR")
require(tree)
# Tree Construnction
tree_accuracy <- tree(formula = Drug ~ Age+Sex+BP+Cholesterol+Na_to_K, data = trainingSet)
#Tree Plot
plot(tree_accuracy)
text(tree_accuracy, pretty = 0)
# Accuracy Calculation
tree_pred = predict(tree_accuracy, drugdata[-train_int,], type = 'class')
with(drugdata[-train_int,], table(tree_pred, Drug))

#Task 4
# Cross-validation and Tree Construction
drug.cv = cv.tree(tree_accuracy, FUN = prune.misclass)
drug.cv > plot(drug.cv)
prune.drug = prune.misclass(tree_accuracy, best = length(tree_accuracy))
# Tree plot
plot(prune.drug)
text(prune.drug, pretty=0)
# Accuracy Calculation
tree_pred = predict(prune.drug, drugdata[-train_int,], type = "class")
with(drugdata[-train_int,], table(tree_pred, Drug))
```

37:1 (Top Level) R Script

Console | Terminal | Jobs

R 4.1.1 · ~/Desktop/University/Second year/T2/Big data/Assignment2/

```
> drug.cv > plot(drug.cv)
logical(0)
> prune.drug = prune.misclass(tree_accuracy, best = length(tree_accuracy))
> # Tree plot
> plot(prune.drug)
> text(prune.drug, pretty=0)
> # Accuracy Calculation
> tree_pred = predict(prune.drug, drugdata[-train_int,], type = "class")
> with(drugdata[-train_int,], table(tree_pred, Drug))
          Drug
tree_pred drugA drugB drugC drugX DrugY
    drugA     6     1     0     0     0
    drugB     0     4     0     0     0
    drugC     0     0     2     0     0
    drugX     0     0     0    12     0
    DrugY     0     0     0     0    15
>
>
```

Environment | History | Connections | Tutorial

Import Dataset · 136 MiB · List

R · Global Environment

Data	
drug.cv	List of 4
drugdata	200 obs. of 6 variables
prune.drug	List of 6
testSet	40 obs. of 6 variables
trainingSet	160 obs. of 6 variables
tree_accuracy	List of 6
Values	
testing_size	40
train_int	int [1:160] 73 57 174 95 186 197 61 188 59 99 ...
training_size	160
tree_pred	Factor w/ 5 levels "drugA","drugB",..: 3 5 1 5 5 5 4 4 4…

Files | Plots | Packages | Help | Viewer

Zoom · Export · Publish