

INFS4203/7203 Data Mining
The University of Queensland, Australia
Semester 2, 2018

Tutorial Week 6:
Introduction to R for Data Mining
(Part II)

Chandra Prasetyo Utomo
c.utomo@uq.edu.au

Objectives

1. To tour some methods in data selection for data mining purposes.
2. To be able to generate and save a plot from data.

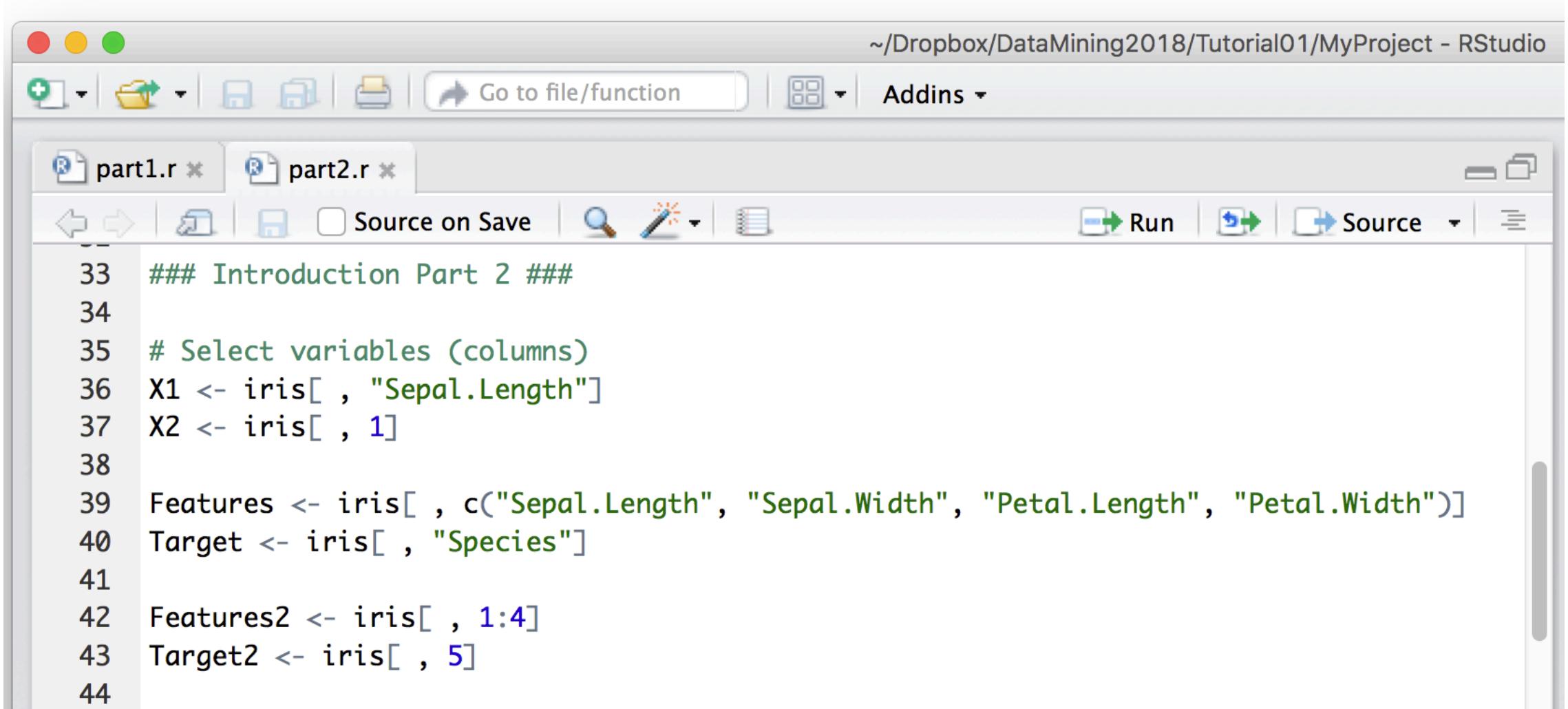
Outline

1. Data Selection (*25 minutes*)
2. Data Visualisation (*25 minutes*)

Part 1

Data Selection

Select Variables (columns)

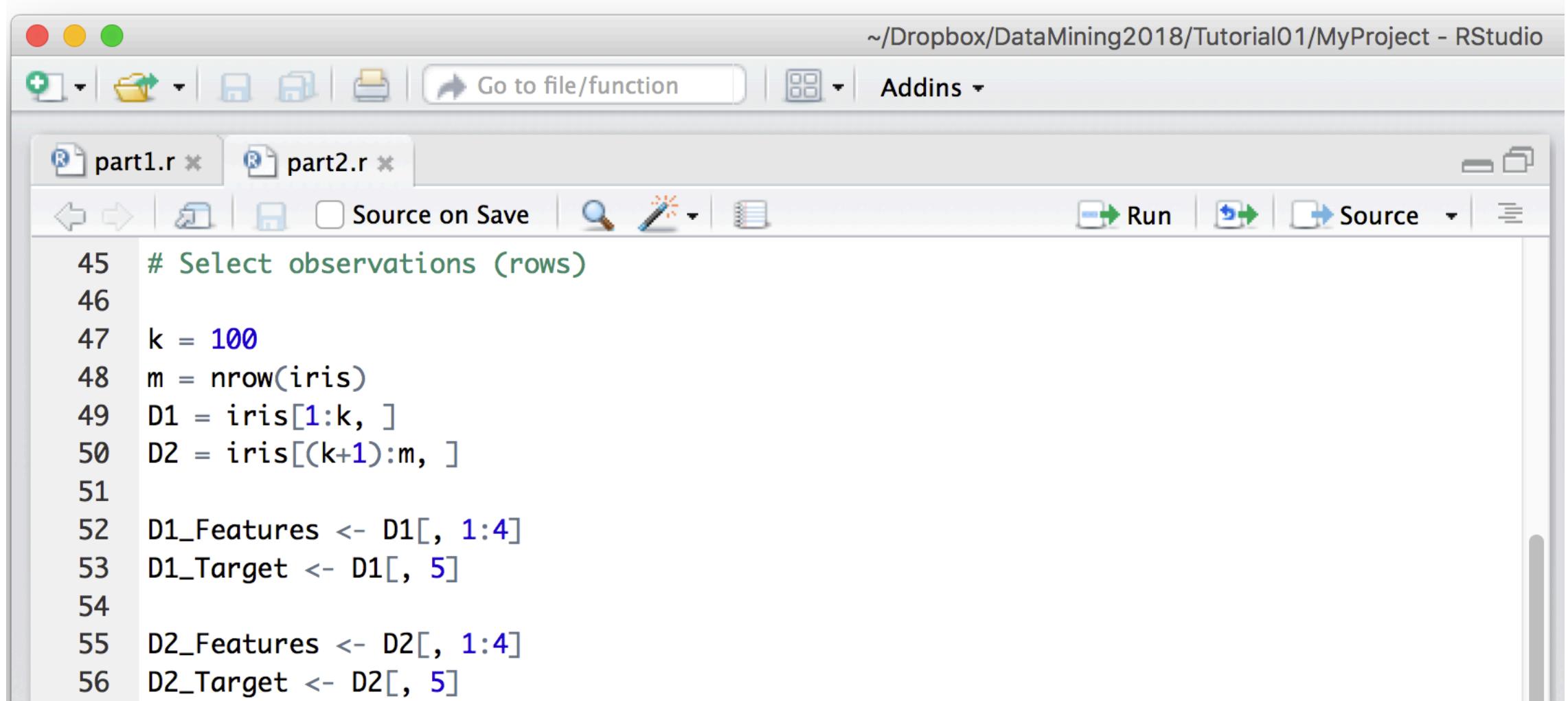


The screenshot shows the RStudio interface with the following details:

- Title Bar:** ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio
- Toolbar:** Includes standard file operations (New, Open, Save, Print, Find, Copy, Paste, Find Next, Find Previous, Go to file/function, Addins).
- File Tab:** Shows two open files: part1.r * and part2.r *.
- Editor Toolbar:** Includes back, forward, search, and edit icons, along with "Source on Save" and "Run" buttons.
- Code Area:** Displays the following R code:

```
33  ### Introduction Part 2 ###
34
35 # Select variables (columns)
36 X1 <- iris[, "Sepal.Length"]
37 X2 <- iris[, 1]
38
39 Features <- iris[, c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width")]
40 Target <- iris[, "Species"]
41
42 Features2 <- iris[, 1:4]
43 Target2 <- iris[, 5]
44
```

Select Observations (rows)



The screenshot shows the RStudio IDE interface. The title bar indicates the project is located at `~/Dropbox/DataMining2018/Tutorial01/MyProject`. The top toolbar includes standard operating system icons (red, yellow, green circles) and application-specific icons for file operations like Open, Save, and Print, along with a "Go to file/function" search bar and an "Addins" dropdown.

The left sidebar displays two open files: `part1.r` and `part2.r`. The main workspace shows the following R code:

```
45 # Select observations (rows)
46
47 k = 100
48 m = nrow(iris)
49 D1 = iris[1:k, ]
50 D2 = iris[(k+1):m, ]
51
52 D1_Features <- D1[, 1:4]
53 D1_Target <- D1[, 5]
54
55 D2_Features <- D2[, 1:4]
56 D2_Target <- D2[, 5]
```

The code uses the `iris` dataset to split it into two parts: `D1` and `D2`, each containing 100 observations. The first 100 rows are assigned to `D1`, and the remaining rows are assigned to `D2`. The features are selected from columns 1 to 4, and the target variable is selected from column 5.

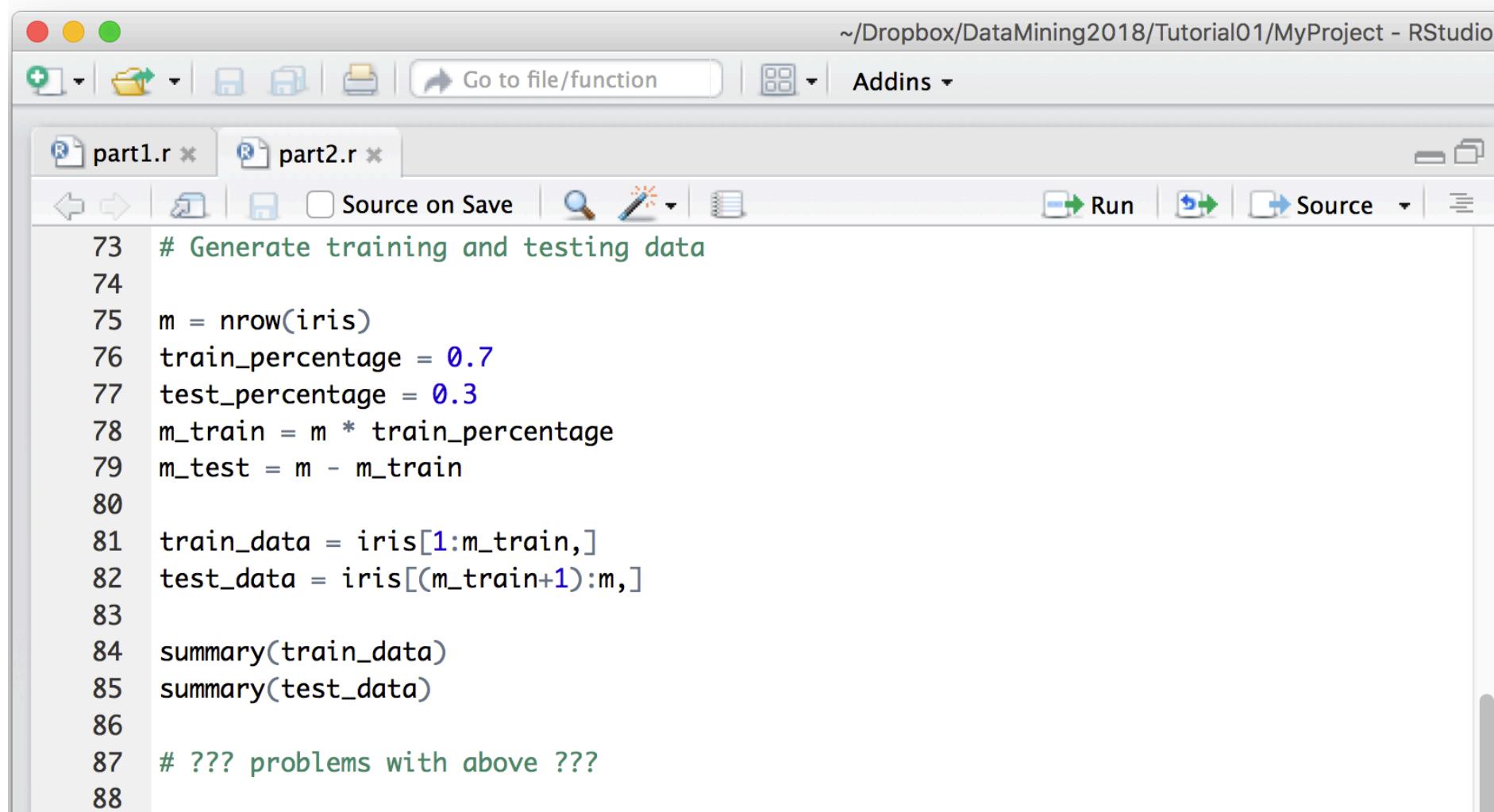
Selection based on Condition(s)

The screenshot shows the RStudio interface with the following details:

- Title Bar:** ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio
- Toolbar:** Includes standard icons for file operations (New, Open, Save, Print), Go to file/function, and Addins.
- File Tab:** Shows two files: part1.r and part2.r.
- Toolbar Buttons:** Source on Save, Run, Source.
- Code Editor:** Displays R code for selecting data based on conditions using the iris dataset.

```
58 # Select based on condition(s)
59
60 iris_setosa <- iris[iris$Species == "Iris-setosa", ]
61 str(iris_setosa)
62
63 high_sepal_length <- iris[iris$Sepal.Length > 5.8, ]
64 str(high_sepal_length)
65 summary(high_sepal_length)
66 summary(high_sepal_length$Sepal.Length)
67
68 low_petal_length <- iris[iris$Petal.Length <= 5.1, ]
69 str(low_petal_length)
70 summary(low_petal_length)
71 summary(low_petal_length$Petal.Length)
```

Generate Training and Testing Data 1

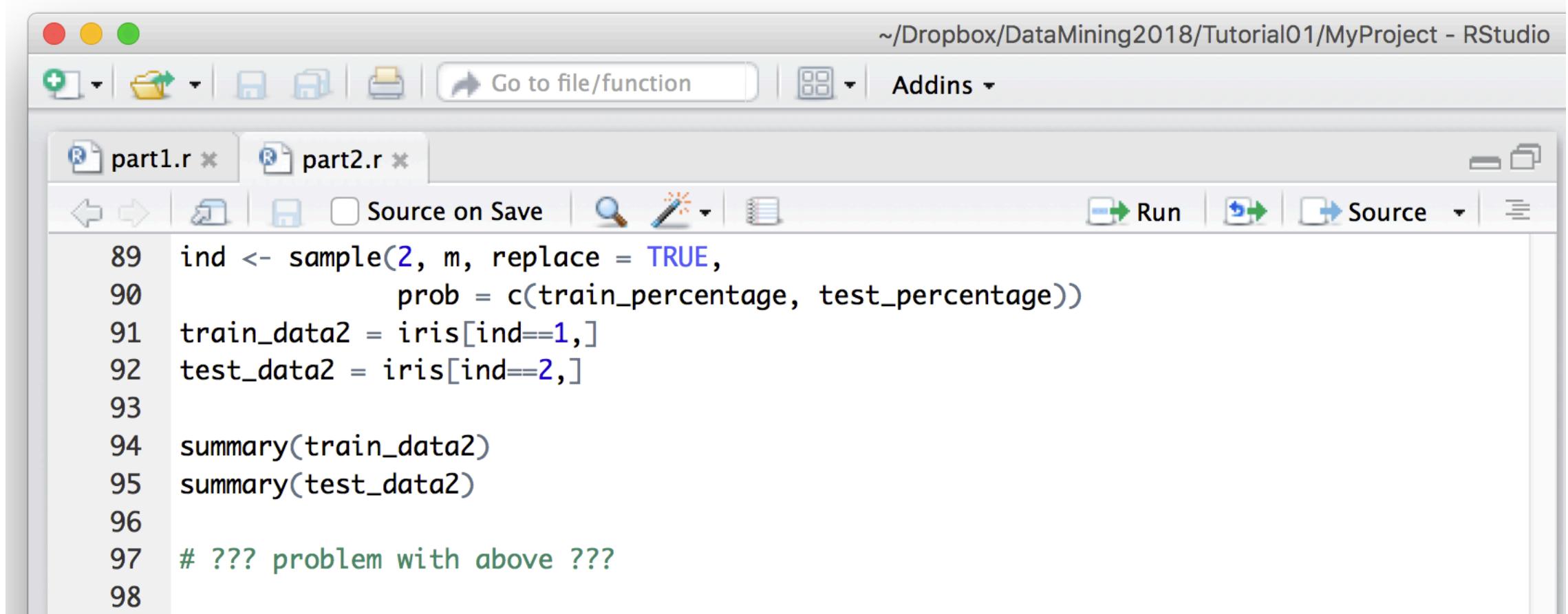


The screenshot shows the RStudio IDE interface. The title bar indicates the project is located at `~/Dropbox/DataMining2018/Tutorial01/MyProject`. The code editor window displays an R script with the following content:

```
73 # Generate training and testing data
74
75 m = nrow(iris)
76 train_percentage = 0.7
77 test_percentage = 0.3
78 m_train = m * train_percentage
79 m_test = m - m_train
80
81 train_data = iris[1:m_train,]
82 test_data = iris[(m_train+1):m,]
83
84 summary(train_data)
85 summary(test_data)
86
87 # ??? problems with above ???
88
```

The code uses the `iris` dataset to calculate the number of rows for training and testing data, then subsets the dataset into `train_data` and `test_data`. It concludes with a note about potential issues with the current approach.

Generate Training and Testing Data 2

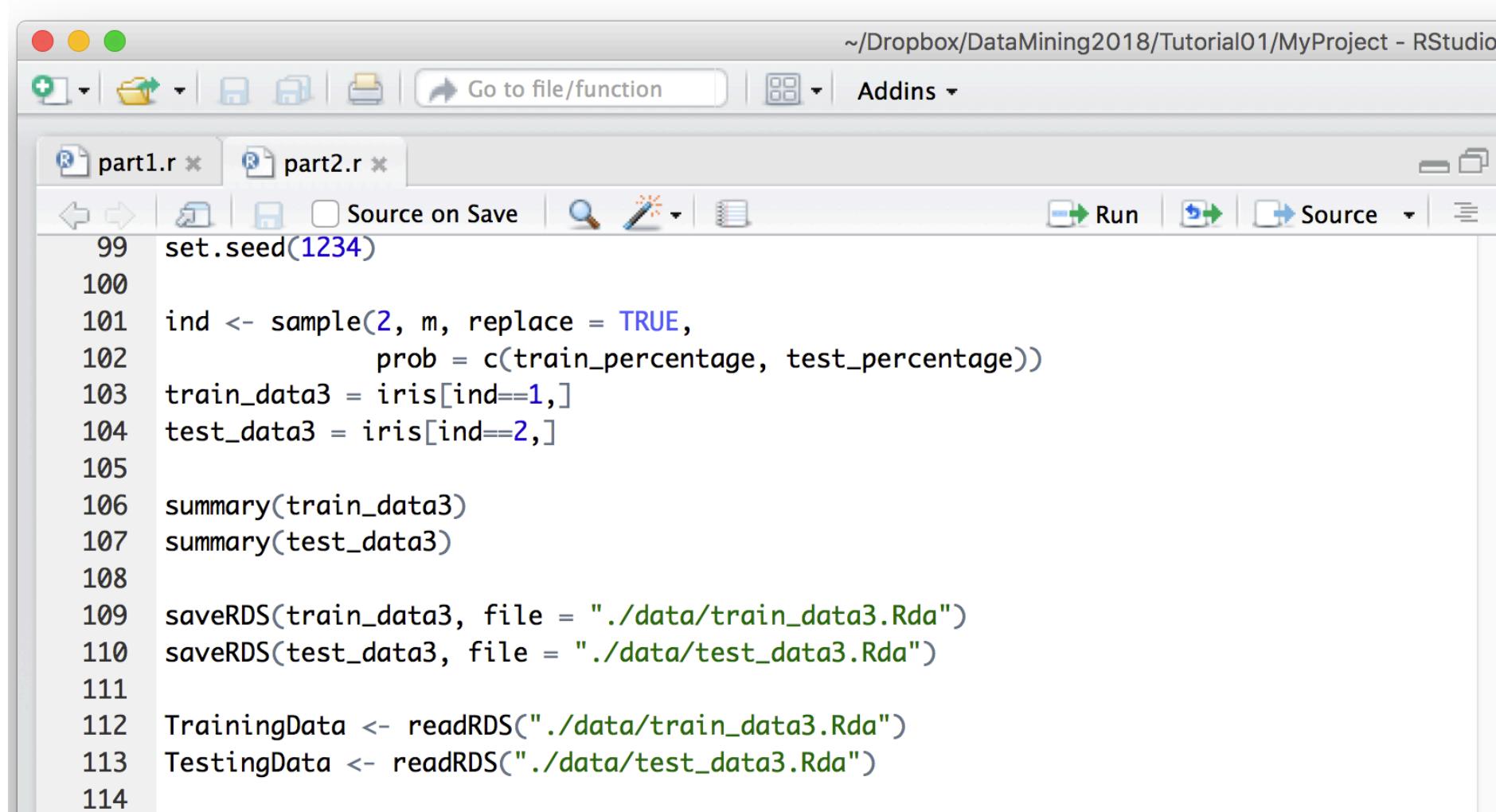


The screenshot shows the RStudio interface with the following details:

- Title Bar:** ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio
- Toolbar:** Includes standard icons for file operations (New, Open, Save, Print), Go to file/function, and Addins.
- File Tab:** Shows two files: part1.r and part2.r, with part1.r currently selected.
- Editor Area:** Displays R code for generating training and testing data from the Iris dataset.
- Code Content:**

```
89 ind <- sample(2, m, replace = TRUE,
90                  prob = c(train_percentage, test_percentage))
91 train_data2 = iris[ind==1,]
92 test_data2 = iris[ind==2,]
93
94 summary(train_data2)
95 summary(test_data2)
96
97 # ??? problem with above ???
98
```
- Toolbars:** Includes Source on Save, Find, Edit, Run, and Source buttons.

Generate Training and Testing Data 3



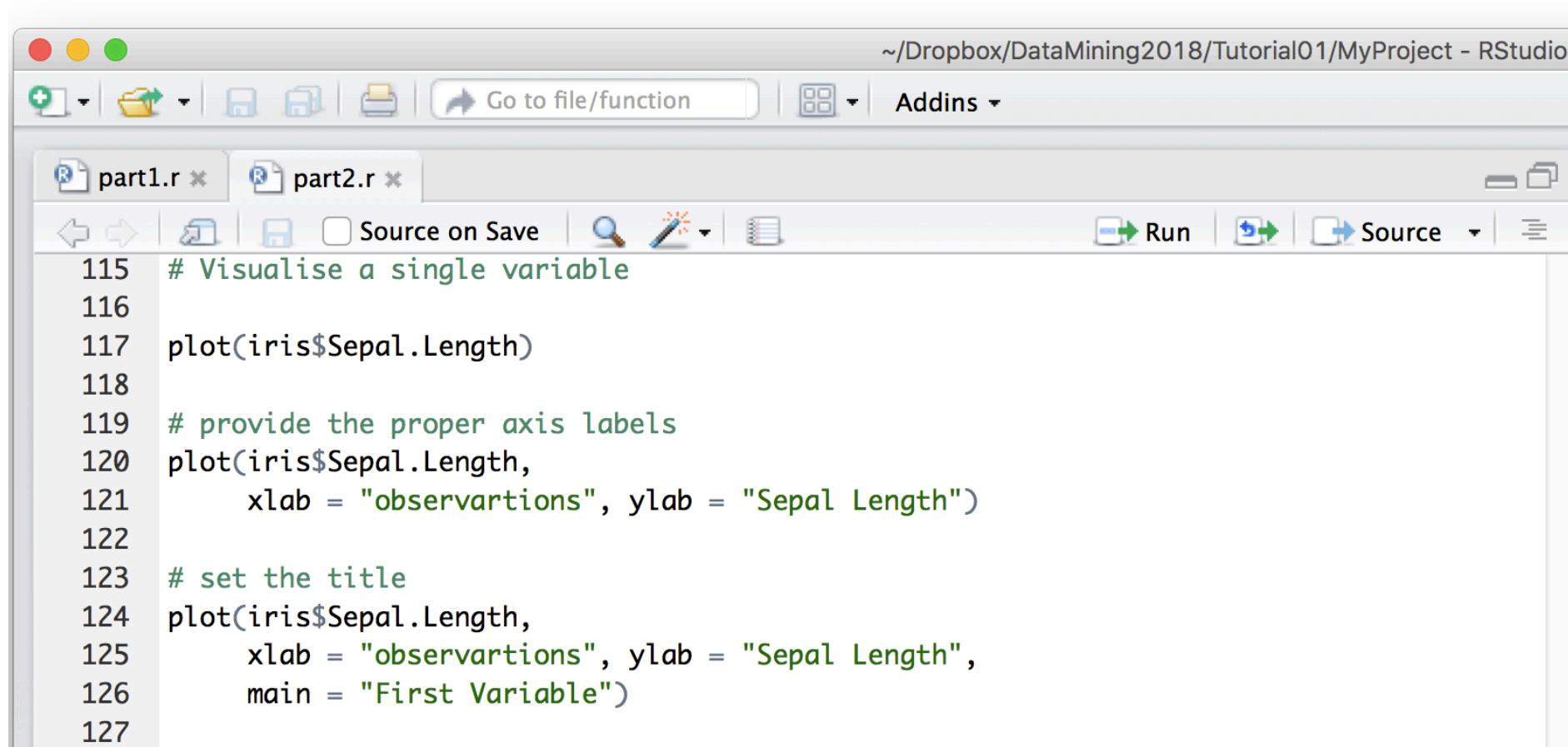
The screenshot shows the RStudio IDE interface. The title bar indicates the project is located at `~/Dropbox/DataMining2018/Tutorial01/MyProject`. The top menu bar includes standard icons for file operations and a "Go to file/function" search bar. Below the menu is a toolbar with various icons for file management and code execution. Two R script files are open in the editor tabs: `part1.r` and `part2.r`. The code in the editor is as follows:

```
99 set.seed(1234)
100
101 ind <- sample(2, m, replace = TRUE,
102                 prob = c(train_percentage, test_percentage))
103 train_data3 = iris[ind==1,]
104 test_data3 = iris[ind==2,]
105
106 summary(train_data3)
107 summary(test_data3)
108
109 saveRDS(train_data3, file = "./data/train_data3.Rda")
110 saveRDS(test_data3, file = "./data/test_data3.Rda")
111
112 TrainingData <- readRDS("./data/train_data3.Rda")
113 TestingData <- readRDS("./data/test_data3.Rda")
114
```

Part 2

Data Visualisation

Visualise a Single Variable

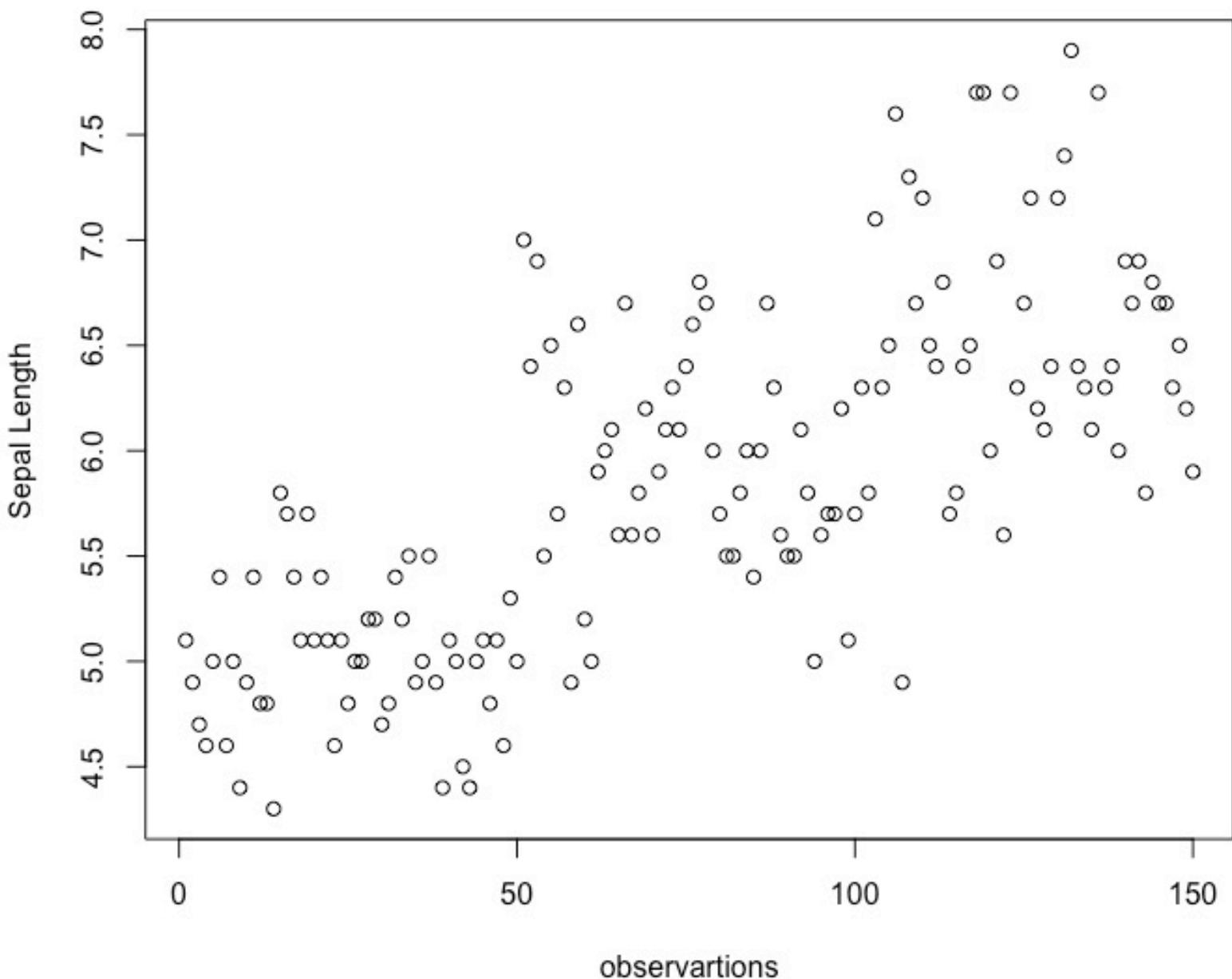


The screenshot shows the RStudio IDE interface. The title bar reads '~Dropbox/DataMining2018/Tutorial01/MyProject - RStudio'. The top menu bar includes standard OS X window controls (red, yellow, green) and application-specific icons for file operations like Open, Save, Print, and Go to file/function. Below the menu is an 'Addins' dropdown. The workspace shows two open files: 'part1.r' and 'part2.r'. The code editor displays the following R script:

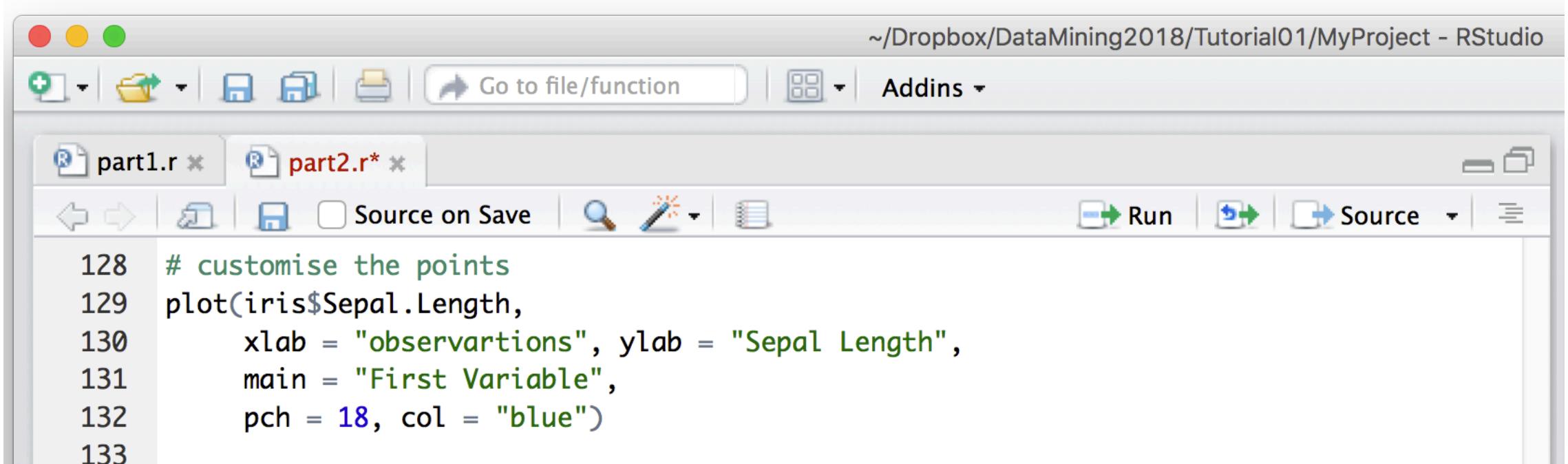
```
115 # Visualise a single variable
116
117 plot(iris$Sepal.Length)
118
119 # provide the proper axis labels
120 plot(iris$Sepal.Length,
121       xlab = "observartions", ylab = "Sepal Length")
122
123 # set the title
124 plot(iris$Sepal.Length,
125       xlab = "observartions", ylab = "Sepal Length",
126       main = "First Variable")
127
```

The code uses the `plot` function to create a scatter plot of Sepal Length for the Iris dataset. It first plots the raw data, then adds axis labels ('observartions' and 'Sepal Length'), and finally adds a title ('First Variable'). The RStudio interface also features a toolbar with icons for back, forward, search, and other functions, along with buttons for Run, Source, and other project management tools.

First Variable



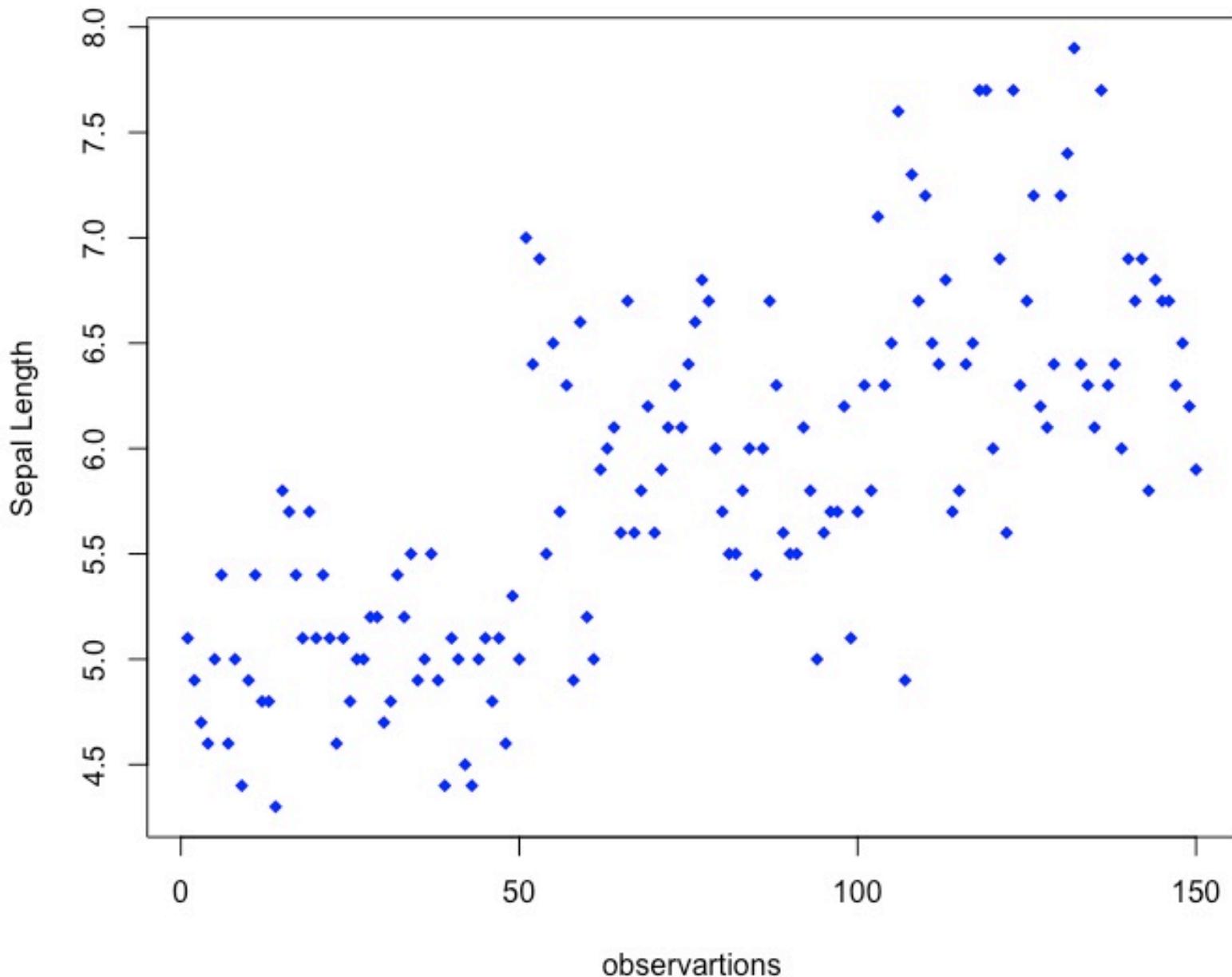
Customise the points



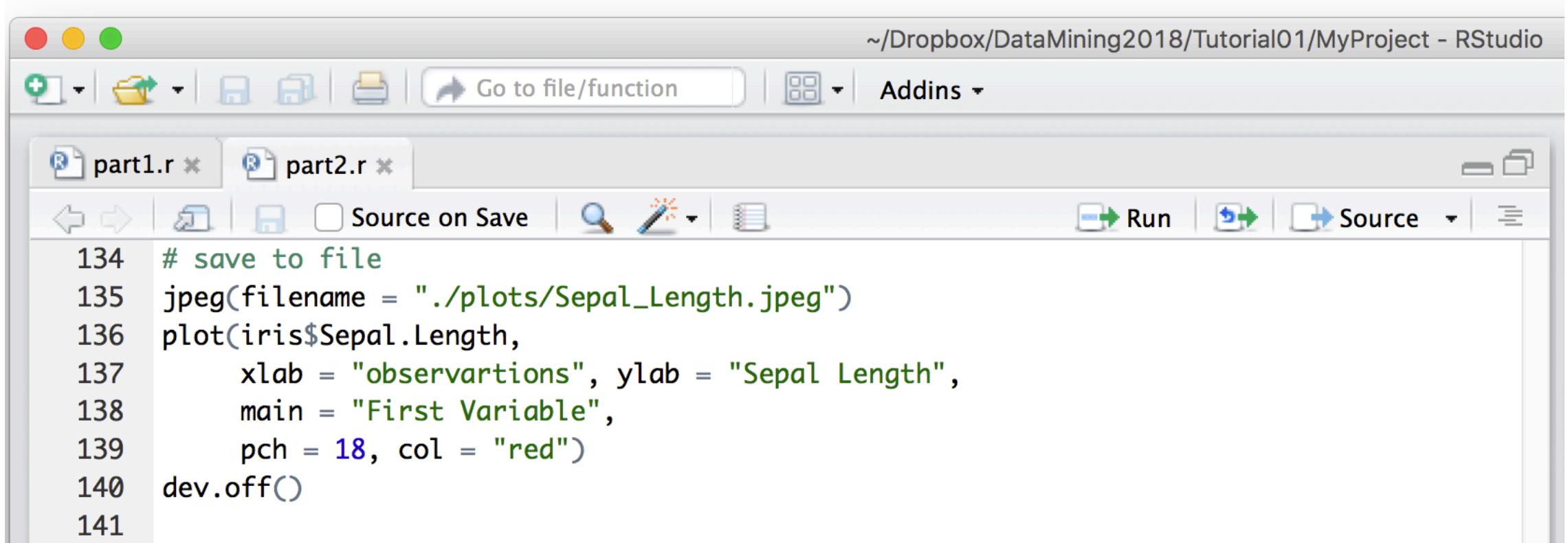
The screenshot shows the RStudio IDE interface. The title bar reads: ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio. The top menu bar includes standard OS X window controls (red, yellow, green buttons), file navigation icons (New, Open, Save, Print, Find, Go to file/function, Addins), and a toolbar with various icons. Below the toolbar, the script editor displays two files: part1.r* and part2.r*. The part1.r* file contains the following R code:

```
128 # customise the points
129 plot(iris$Sepal.Length,
130       xlab = "observartions", ylab = "Sepal Length",
131       main = "First Variable",
132       pch = 18, col = "blue")
133
```

First Variable



Save to File

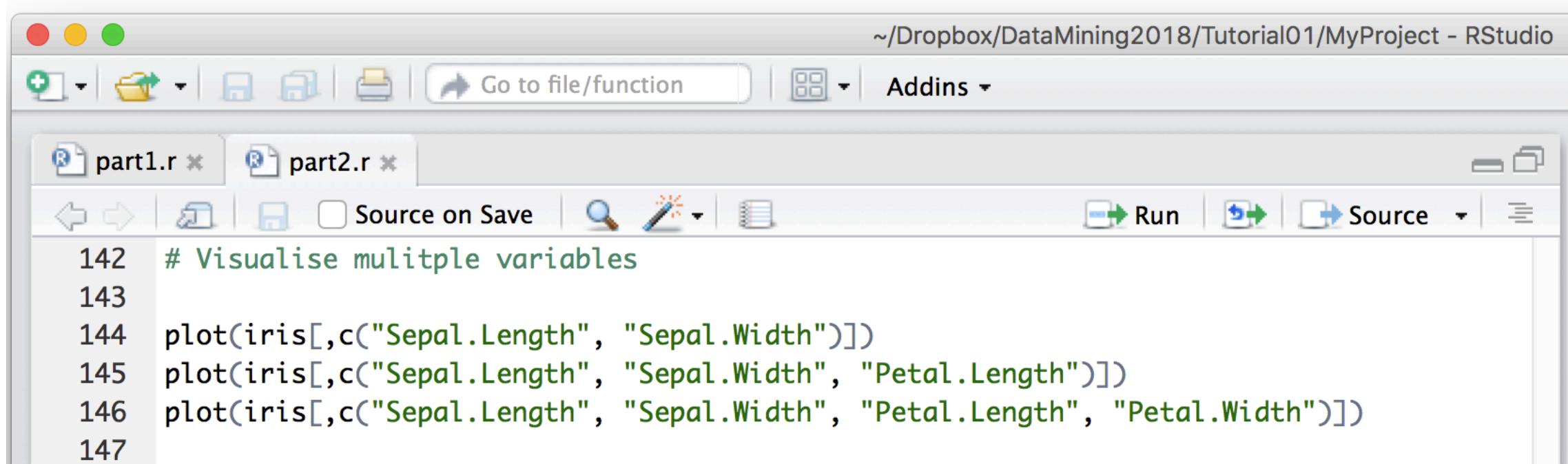


The screenshot shows the RStudio IDE interface. The title bar reads " ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio". The toolbar includes standard Mac OS X window controls, a file menu icon, a folder icon, a search icon, and a "Go to file/function" button. To the right of the search button is an "Addins" dropdown. Below the toolbar, two files are listed in the project tree: "part1.r" and "part2.r". The main workspace displays the following R code:

```
134 # save to file
135 jpeg(filename = "./plots/Sepal_Length.jpeg")
136 plot(iris$Sepal.Length,
137       xlab = "observartions", ylab = "Sepal Length",
138       main = "First Variable",
139       pch = 18, col = "red")
140 dev.off()
141
```

The code uses the `jpeg` function to save a plot of Sepal Length to a file named "Sepal_Length.jpeg". The plot is created using the `plot` function with specific parameters for x-axis label, y-axis label, main title, and point character.

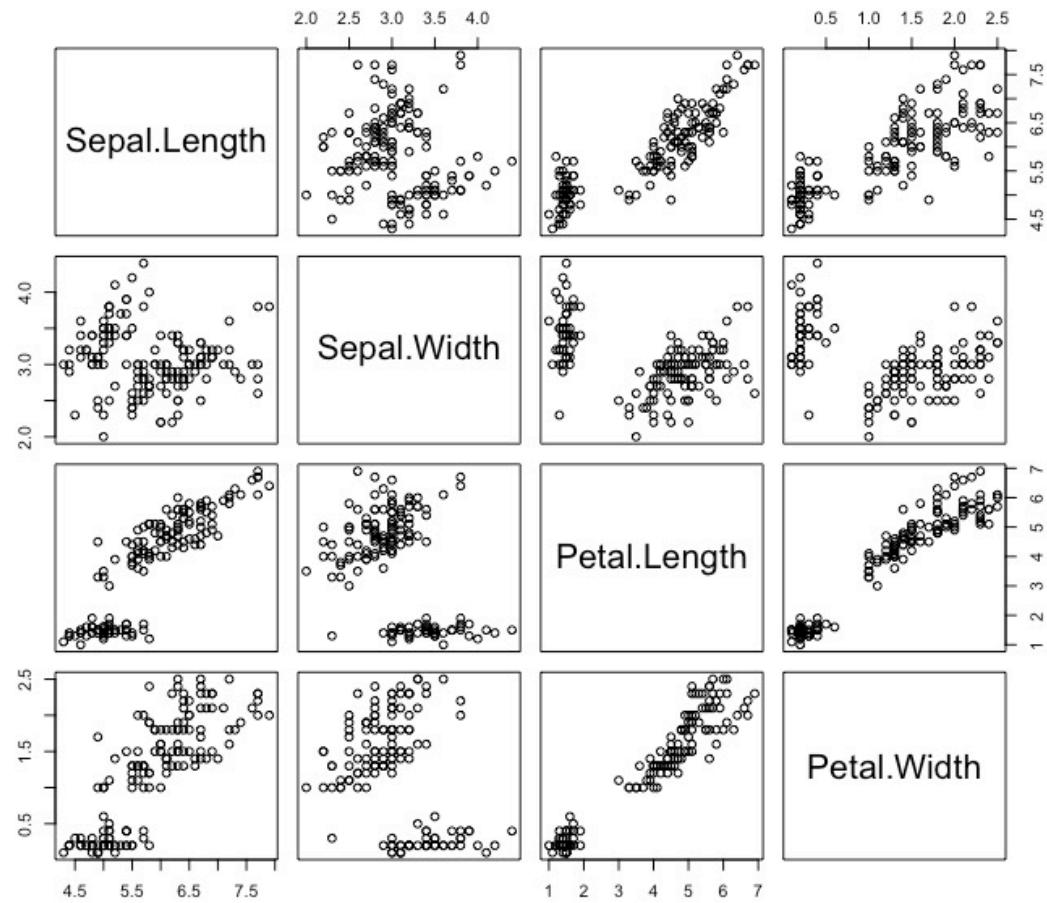
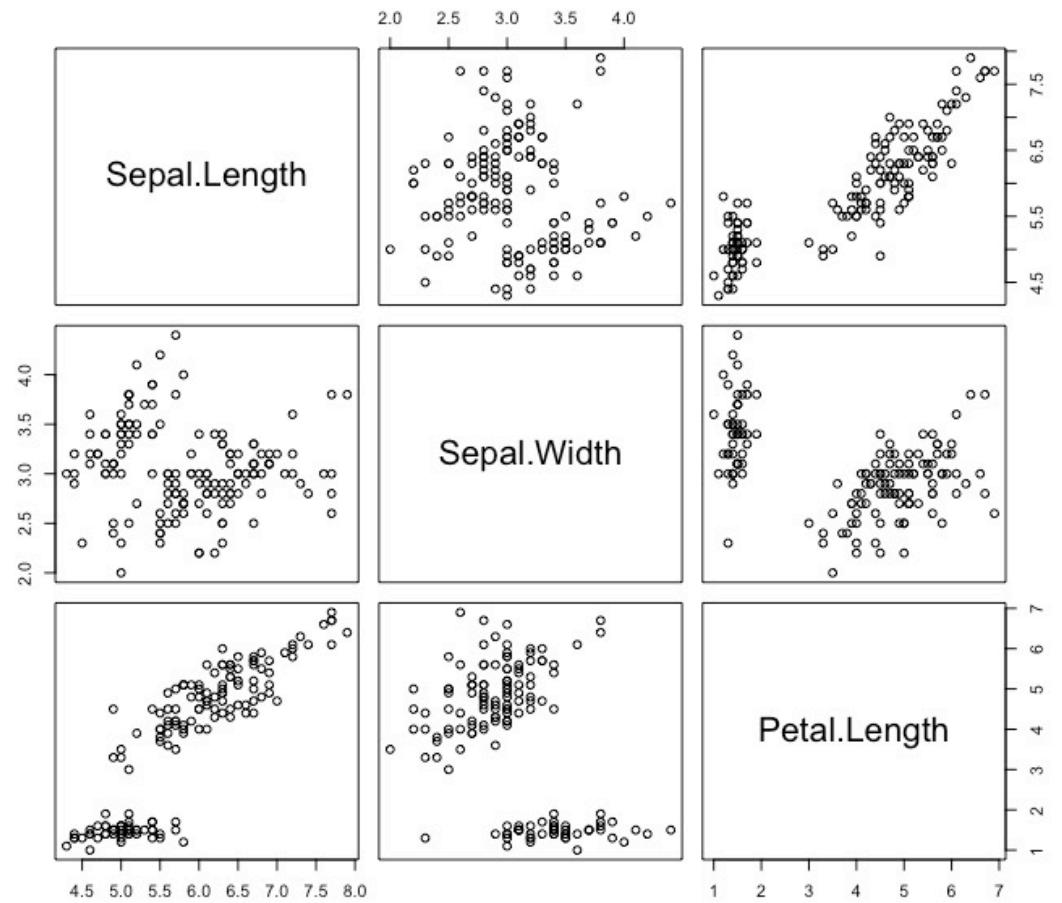
Visualise Multiple Variables



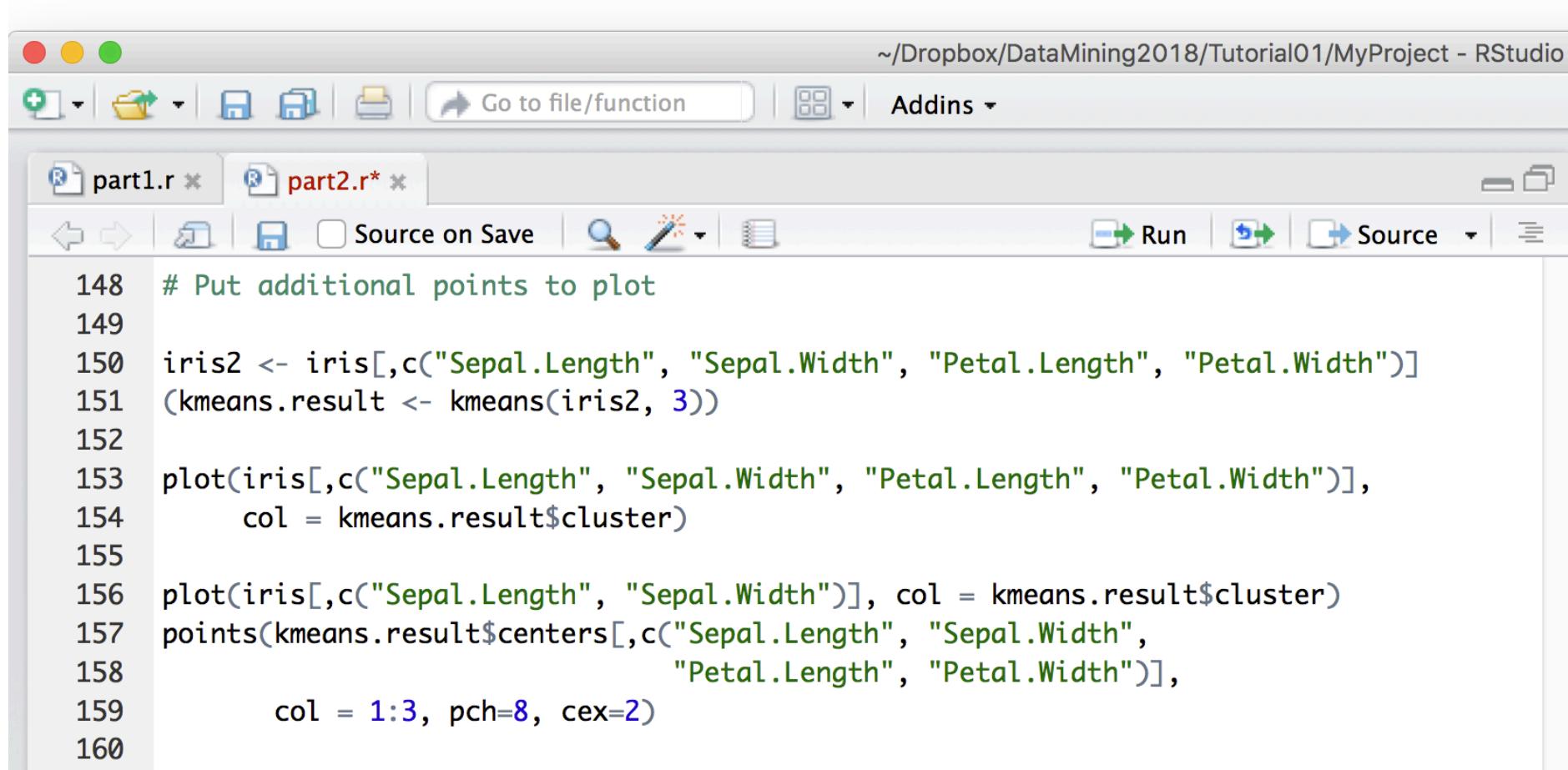
The screenshot shows the RStudio IDE interface. The title bar indicates the project path: ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio. The toolbar includes standard icons for file operations like Open, Save, and Print, along with a 'Go to file/function' search bar and an 'Addins' dropdown. Below the toolbar, two R script files are listed: 'part1.r' and 'part2.r'. The main editor area displays the following R code:

```
142 # Visualise mulitple variables
143
144 plot(iris[,c("Sepal.Length", "Sepal.Width")])
145 plot(iris[,c("Sepal.Length", "Sepal.Width", "Petal.Length")])
146 plot(iris[,c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width")])
147
```

The code uses the `plot` function to create scatter plots for different combinations of variables from the Iris dataset.



Add points to plot



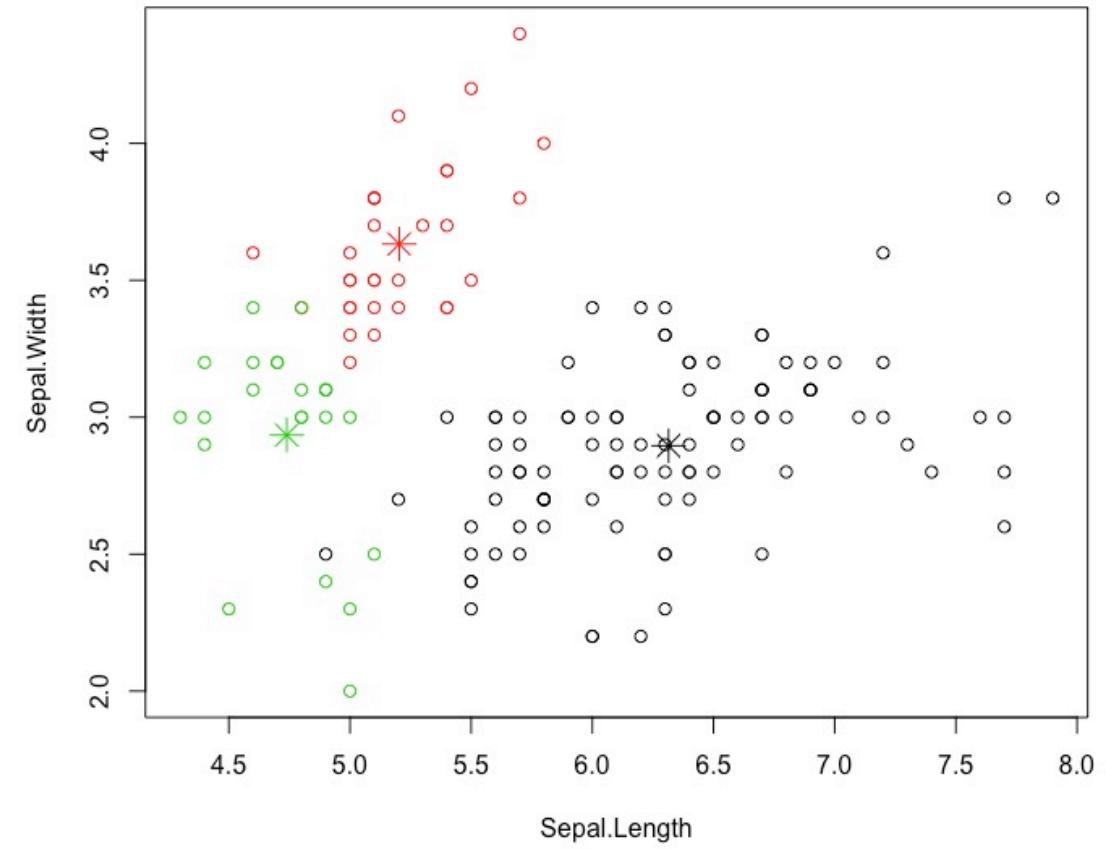
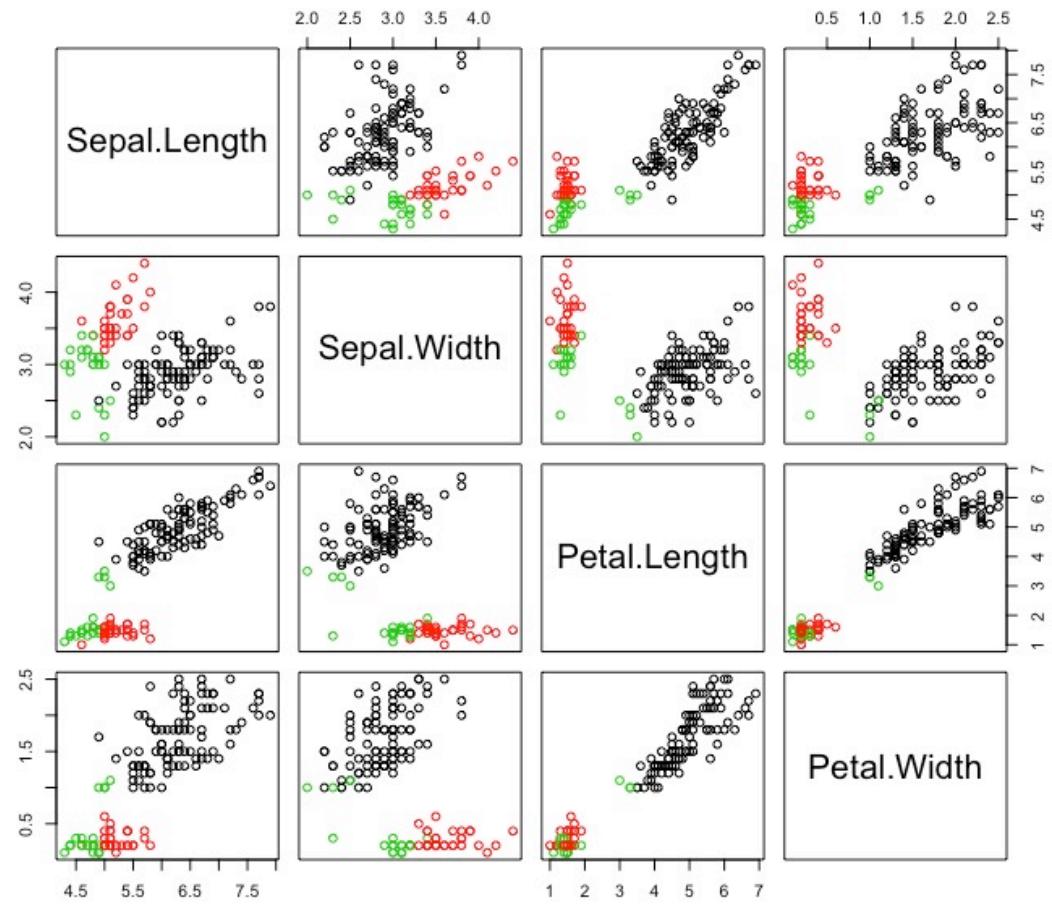
The screenshot shows the RStudio interface with the following details:

- Title Bar:** ~/Dropbox/DataMining2018/Tutorial01/MyProject - RStudio
- Toolbar:** Includes standard icons for file operations (New, Open, Save, Print) and navigation (Go to file/function).
- File Tab:** Shows two files: part1.r and part2.r*.
- Editor Area:** Displays the following R code:

```
148 # Put additional points to plot
149
150 iris2 <- iris[,c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width")]
151 (kmeans.result <- kmeans(iris2, 3))
152
153 plot(iris[,c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width")],
154       col = kmeans.result$cluster)
155
156 plot(iris[,c("Sepal.Length", "Sepal.Width")], col = kmeans.result$cluster)
157 points(kmeans.result$centers[,c("Sepal.Length", "Sepal.Width",
158                               "Petal.Length", "Petal.Width")],
159           col = 1:3, pch=8, cex=2)
160
```

Don't worry about k-means clustering at the moment.

We will go deeper and discuss more detail about it in the next tutorial.



Reference:

- R and Data Mining. Yangchan Zhao. Academic Press 2012.
<https://www.sciencedirect.com/book/9780123969637/r-and-data-mining>