

Complete R Programming for Data Science Cheatsheet

Author: Learn Everything AI - Shivam Modi

Introduction

This cheatsheet provides a comprehensive overview of R programming concepts for data science beginners. It covers essential topics and provides code examples to demonstrate the concepts.

Data Types

- Numeric: 1, 2.5, -3.14
- Character: "Hello", "World"
- Logical: TRUE, FALSE
- Integer: 1L, 2L, -3L
- Complex: 2 + 3i, -1.5 - 2i
- Vector: c(1, 2, 3)
- Matrix: matrix(data, nrow, ncol)

Control Structures

- if-else:

```
1 if (condition) {  
2   # Code block executed if condition is true  
3 } else {  
4   # Code block executed if condition is false  
5 }
```

- for loop:

```
1 for (variable in sequence) {  
2   # Code block executed for each element  
3 }
```

- while loop:

```
1 while (condition) {  
2   # Code block executed while condition is true  
3 }  
4 \item \texttt{repeat} loop:  
5 \begin{lstlisting}[style=mystyle]  
6 repeat {  
7   # Code block executed indefinitely  
8   # Use 'break' statement to exit the loop  
9 }
```

Functions

- Defining a function:

```
1 my_function <- function(arg1, arg2) {  
2   # Code block defining the function  
3   return(result)  
4 }
```

- Calling a function:

```
1 result <- my_function(value1, value2)
```

Data Manipulation

- Data frame creation:

```
1 df <- data.frame(column1, column2, ...)
```

- Selecting columns:

```
1 selected_columns <- df[, c("column1", "column2")]
```

- Filtering rows:

```
1 filtered_rows <- df[df$column > 5, ]
```

- Sorting data:

```
1 sorted_data <- df[order(df$column), ]
```

- Aggregating data:

```
1 aggregated_data <- aggregate(column ~ group, data=df, FUN=sum)
```

Data Visualization

- Histogram:

```
1 hist(data, main="Histogram", xlab="Values")
```

- Scatter plot:

```
1 plot(x, y, main="Scatter Plot", xlab="X", ylab="Y")
```

- Bar plot:

```
1 barplot(heights, main="Bar Plot", xlab="Categories", ylab="Counts")
```

Complete R Programming for Data Mining Cheatsheet

Introduction

This cheatsheet provides a comprehensive overview of R programming concepts for data mining. It is designed to assist beginners in understanding and implementing data mining techniques using R.

Installation

To get started with R and data mining packages, follow these steps:

```
1 # Install R
2 install.packages("r-base")
3
4 # Install required packages
5 install.packages(c("tidyverse", "caret", "arules"))
```

Data Manipulation

R provides powerful tools for data manipulation and preprocessing.

Loading Data

To load a dataset in R, use the following code:

```
1 # Load CSV data
2 data <- read.csv("data.csv")
3
4 # Load Excel data
5 library(readxl)
6 data <- read_excel("data.xlsx")
```

Data Exploration

R offers various functions for exploring and summarizing data.

```
1 # Summary statistics
2 summary(data)
3
4 # Correlation matrix
5 cor(data)
6
7 # Frequency table
8 table(data$column)
```

Data Mining Techniques

Association Rule Mining

To perform association rule mining in R:

```

1 # Load the arules package
2 library(arules)
3
4 # Convert data to transaction format
5 transactions <- as(data, "transactions")
6
7 # Generate frequent itemsets
8 frequent_itemsets <- eclat(transactions, parameter = list(support = 0.1))
9
10 # Generate association rules
11 association_rules <- apriori(transactions, parameter = list(support = 0.1, confidence = 0.5)
12 )
13 # Explore association rules
14 inspect(association_rules)

```

Clustering

To perform clustering analysis in R:

```

1 # Load the cluster package
2 library(cluster)
3
4 # Perform k-means clustering
5 kmeans_result <- kmeans(data, centers = 3)
6
7 # Explore clustering results
8 kmeans_result$cluster

```

Classification

To perform classification in R:

```

1 # Load the caret package
2 library(caret)
3
4 # Split data into training and testing sets
5 set.seed(123)
6 trainIndex <- createDataPartition(data$target, p = 0.8, list = FALSE)
7 trainData <- data[trainIndex, ]
8 testData <- data[-trainIndex, ]
9
10 # Train a classification model
11 classification_model <- train(target ~ ., data = trainData, method = "svm")
12
13 # Make predictions on test data
14 predictions <- predict(classification_model, newdata = testData)

```

Text Mining

To perform text mining in R:

```

1 # Load the tm package
2 library(tm)
3
4 # Create a corpus from text data
5 corpus <- Corpus(VectorSource(text_data))
6
7 # Preprocess the corpus
8 corpus <- tm_map(corpus, content_transformer(tolower))
9 corpus <- tm_map(corpus, removeNumbers)
10 corpus <- tm_map(corpus, removePunctuation)
11 corpus <- tm_map(corpus, removeWords, stopwords("english"))
12 corpus <- tm_map(corpus, stripWhitespace)
13

```

```
14 # Create a document-term matrix
15 dtm <- DocumentTermMatrix(corpus)
16
17 # Perform text mining tasks (e.g., topic modeling, sentiment analysis)
```

Model Evaluation

To evaluate data mining models in R:

```
1 # Load the caret package
2 library(caret)
3
4 # Evaluate the model using cross-validation
5 set.seed(123)
6 model <- train(target ~ ., data = trainData, method = "svm")
7 cv_results <- trainControl(method = "cv", number = 5)
8 cv <- train(model, trControl = cv_results)
```

Conclusion

This cheatsheet provides a comprehensive overview of R programming concepts for data mining. It covers data manipulation, loading data, data exploration, and various data mining techniques such as association rule mining, clustering, classification, and text mining. By following this cheatsheet, beginners can gain a solid understanding of R's capabilities for data mining tasks.

Author

Learn Everything AI - Shivam Modi