Chapter 12
Data Mining for Junior Data Scientists:
Data Analytics With Python

ABSTRACT

It is crucial for junior data scientists to learn computer programming as data science software packages may not always cater to the requirements of data analysis. Python provides a vast library of algorithms for data analysis, including NumPy, Pandas, Matplotlib, Seaborn, and Scikit-learn. NumPy and Pandas aid in organizing datasets as part of the pre-processing stage, while Matplotlib and Seaborn offer a range of data visualization commands. These visualization tools are instrumental in data exploration processes, such as creating histograms and scatter plots, and displaying data mining results like cluster analysis outcomes. Scikit-learn is a popular library in the data science industry that offers various data mining commands for regression, decision constructs, and cluster analysis, covering both supervised and unsupervised learning. Therefore, junior data scientists must learn Python programming for data science applications, especially when using software packages that require editing the model using Python commands.

INTRODUCTION

Nowadays, there are tools used for analyzing data during the workflow; the data extraction tools, data survey tools, data preparation tools, Data Analysis Tools with Data Mining Techniques and Data Visualization tools. Each step has a ready-to-use software in both instant software and language programming software. Python is a tool that supports the entire workflow as it can be used for data analysis purposes; manipulating datasets, importing routines, developing data visualization, and data analysis with data mining techniques using libraries (Massaron & Mueller, 2015: Mueller & Massaron, 2019).

DOI: 10.4018/978-1-6684-4730-7.ch012

NumPy Library

The NumPy library provides a set of commands that data scientists can execute to manipulate imported datasets and then process them within Python's command set. The library focuses on compiling data into the format ready to process with data mining techniques:

Array

Python supports storing data in arrays, which are in rows and columns. Array variables can store data of one to more dimensions. The structure is as follows:

Table 1. Dimension array

	Column 0
Row 0	0,0
Row 1	1,0
Row 2	2,0
Row.	.
Row.	.
Row n	n,0

From the table, data scientists can store data as a 1-dimensional array with records starting from 1 row and 1 column. Inside array variables, the location of the data is specified at row 0 and columns 0. For example, the number 2 might be contained in row 0 and column 0. When data scientists store more than 1 column, a 2-dimensional array variable can be created with the following structure:

Table 2. Dimension array

	Column 0	**Column 1**	**Column 2**	**Column .**	**Column .**	**Column n**
Row 0	0,0	0,1	0,2	0,.	0,.	0,n
Row 1	1,0	1,1	1,2	1,.	1,.	1,n
Row 2	2,0	2,1	2,2	2,.	2,.	2,n
Row .	.,0	.,1	.,2	.,.	.,.	.,n
Row .	.,0	.,1	.,2	.,.	.,.	.,n
Row n	n,0	n,1	n,2	n,.	n,.	n,n

From the table, data scientists can store data in multiple rows and columns simultaneously. For example, the numeric data 25 may be positioned in Row 5, Column 3. By storing data in this manner, the data scientists can design data structures with good memory management.