

# Module-03, Python for Data Analysis

## Data Exploration (Pandas-Part-01)

Dostdar Ali  
Instructor

Data science and Artificial Intelligence  
3-Months Course  
at  
Karakaroum international Univrsity

December 28, 2023



# Table of Contents

- 1 Introduction to Pandas
- 2 Series
- 3 DataFrames
- 4 Missing Data
- 5 GroupBy
- 6 Merging, Joining and Concatenating
- 7 Operations
- 8 Data Input and Output



# Introduction to Pandas

- Pandas provides high-level data structures and functions designed to make working with structured or tabular data fast, easy, and expressive. Since its emergence in 2010, it has helped enable Python to be a powerful and productive data analysis environment.
- The primary objects in pandas that will be used in this lectures are the DataFrame, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object
- In this section of the course we will learn how to use pandas for data analysis. You can think of pandas as an extremely powerful version of Excel, with a lot more features.
- I will covered in this lecture some of them.



# Introduction to Pandas

- Pandas provides high-level data structures and functions designed to make working with structured or tabular data fast, easy, and expressive. Since its emergence in 2010, it has helped enable Python to be a powerful and productive data analysis environment.
- The primary objects in pandas that will be used in this lectures are the DataFrame, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object
- In this section of the course we will learn how to use pandas for data analysis. You can think of pandas as an extremely powerful version of Excel, with a lot more features.
- I will covered in this lecture some of them.



# Introduction to Pandas

- Pandas provides high-level data structures and functions designed to make working with structured or tabular data fast, easy, and expressive. Since its emergence in 2010, it has helped enable Python to be a powerful and productive data analysis environment.
- The primary objects in pandas that will be used in this lectures are the DataFrame, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object
- In this section of the course we will learn how to use pandas for data analysis. You can think of pandas as an extremely powerful version of Excel, with a lot more features.
- I will covered in this lecture some of them.



# Introduction to Pandas

- Pandas provides high-level data structures and functions designed to make working with structured or tabular data fast, easy, and expressive. Since its emergence in 2010, it has helped enable Python to be a powerful and productive data analysis environment.
- The primary objects in pandas that will be used in this lectures are the DataFrame, a tabular, column-oriented data structure with both row and column labels, and the Series, a one-dimensional labeled array object
- In this section of the course we will learn how to use pandas for data analysis. You can think of pandas as an extremely powerful version of Excel, with a lot more features.
- I will covered in this lecture some of them.



# Topics to Covered

Pandas blends the high-performance, array-computing ideas of NumPy with the flexible data manipulation capabilities of spreadsheets and relational databases (such as SQL). It provides sophisticated indexing functionality to make it easy to reshape, slice and dice, perform aggregations, and select subsets of data. Since data manipulation. In this Module of the lecture, you should go through the notebooks in this order:

- Series
- DataFrames
- Missing Data



The first main data type we will learn about for pandas is the Series data type. Let's import Pandas and explore the Series object.

A Series is very similar to a NumPy array (in fact it is built on top of the NumPy array object). What differentiates the NumPy array from a Series, is that a Series can have axis labels, meaning it can be indexed by a label, instead of just a number location. It also doesn't need to hold numeric data, it can hold any arbitrary Python Object.

- Creating a Series

You can convert a list, numpy array, or dictionary to a Series.

- Data in a Series

A pandas Series can hold a variety of object types.

- Using an Index

The key to using a Series is understanding its index. Pandas makes use of these index names or numbers by allowing for fast look ups of information (works like a hash table or dictionary).





The first main data type we will learn about for pandas is the Series data type. Let's import Pandas and explore the Series object.

A Series is very similar to a NumPy array (in fact it is built on top of the NumPy array object). What differentiates the NumPy array from a Series, is that a Series can have axis labels, meaning it can be indexed by a label, instead of just a number location. It also doesn't need to hold numeric data, it can hold any arbitrary Python Object.

- Creating a Series

You can convert a list, numpy array, or dictionary to a Series.

- Data in a Series

A pandas Series can hold a variety of object types.

- Using an Index

The key to using a Series is understanding its index. Pandas makes use of these index names or numbers by allowing for fast look ups of information (works like a hash table or dictionary).



The first main data type we will learn about for pandas is the Series data type. Let's import Pandas and explore the Series object.

A Series is very similar to a NumPy array (in fact it is built on top of the NumPy array object). What differentiates the NumPy array from a Series, is that a Series can have axis labels, meaning it can be indexed by a label, instead of just a number location. It also doesn't need to hold numeric data, it can hold any arbitrary Python Object.

- Creating a Series

You can convert a list, numpy array, or dictionary to a Series.

- Data in a Series

A pandas Series can hold a variety of object types.

- Using an Index

The key to using a Series is understanding its index. Pandas makes use of these index names or numbers by allowing for fast look ups of information (works like a hash table or dictionary).



DataFrames are the workhorse of pandas and are directly inspired by the R programming language. We can think of a DataFrame as a bunch of Series objects put together to share the same index.

- Selection and Indexing

Let's learn the various methods to grab data from a DataFrame

- Pass a list of column names.
- Dot method to called it.
- How to apply different methods in this DataFrame, like adding new columns, dropping, iloc and loc etc.



# DataFrames

DataFrames are the workhorse of pandas and are directly inspired by the R programming language. We can think of a DataFrame as a bunch of Series objects put together to share the same index.

- Selection and Indexing

Let's learn the various methods to grab data from a DataFrame

- Pass a list of column names.
- Dot method to called it.
- How to apply different methods in this DataFrame, like adding new columns, dropping, iloc and loc etc.



# DataFrames

DataFrames are the workhorse of pandas and are directly inspired by the R programming language. We can think of a DataFrame as a bunch of Series objects put together to share the same index.

- Selection and Indexing

Let's learn the various methods to grab data from a DataFrame

- Pass a list of column names.
- Dot method to called it.
- How to apply different methods in this DataFrame, like adding new columns, dropping, iloc and loc etc.



# DataFrames

DataFrames are the workhorse of pandas and are directly inspired by the R programming language. We can think of a DataFrame as a bunch of Series objects put together to share the same index.

- Selection and Indexing

Let's learn the various methods to grab data from a DataFrame

- Pass a list of column names.
- Dot method to called it.
- How to apply different methods in this DataFrame, like adding new columns, dropping, iloc and loc etc.



# Missing Data

Let's show a few convenient methods to deal with Missing Data in pandas:

- dropna method with along row or column and thresh setting.
- fillna method with mean, mode, and average etc.



content...





# Merging, Joining and Concatenating

content...



content...



# Data Input and Output

content...



Great Job  
Thank yo

