Iris Dataset Analysis Using Pandas

May 8, 2025

Contents

1 Introduction

This document explains a data analysis project using the Iris dataset, a classic dataset in data science, to demonstrate various data manipulation techniques with the pandas library in Python. The analysis includes filtering, merging, sorting, transposing, melting, and pivoting the dataset to explore its structure and properties. The Iris dataset is widely used for educational purposes due to its simplicity and clear structure.

The objective is to perform a series of data transformations and explain each step clearly for an oral presentation. This document is designed to provide a comprehensive yet concise overview to aid in understanding and presenting the analysis effectively.

2 Dataset Overview

The Iris dataset contains 150 records of iris flowers, each described by 5 columns: 4 numeric features and 1 categorical target variable. The columns are:

- `sepal.length`: Sepal length in centimeters
- `sepal.width`: Sepal width in centimeters
- `petal.length`: Petal length in centimeters
- `petal.width`: Petal width in centimeters
- `variety`: Iris species (Setosa, Versicolor, or Virginica)

Each species has 50 records, making the dataset balanced. The goal is to manipulate this dataset using pandas to explore its properties and demonstrate data handling techniques.

3 Data Analysis Steps

The analysis involves several pandas operations, each serving a specific purpose in data exploration or transformation. Below is a detailed explanation of each step.

3.1 Filtering by Variety

The dataset was filtered to create subsets for each iris species:

- `Setosa`: Selected rows where `variety == 'Setosa'`, resulting in 50 rows.
- `Versicolor`: Selected rows where `variety == 'Versicolor'`, resulting in 50 rows.
- `Virginica`: Selected rows where `variety == 'Virginica'`, resulting in 50 rows.

The Setosa subset, for example, contains 50 rows with measurements for Setosa flowers, such as sepal length ranging from 4.3 to 5.8 cm.

3.2 Merging Subsets

The Setosa and Versicolor subsets were merged using pd.concat([satosa, versicolor]) to create a combined dataset with 100 rows (50 Setosa + 50 Versicolor). This operation demonstrates how to combine datasets vertically, preserving all columns and stacking the rows.

3.3 Sorting by Petal Length

The dataset was sorted by petal.length in descending order using df.sort$_values(by =' petal.length', ascending = False).Theresultingdataset(150rows)listsflowerswiththelargestpetalleng$

3.4 Transposing the Dataset

The dataset was transposed using df.transpose(), converting rows to columns and columns to rows. The transposed dataset has 5 rows (corresponding to the original columns: sepal.length, sepal.width, petal.length, petal.width, variety) and 150 columns (one for each flower). This operation is useful for changing the dataset’s perspective, though it’s less common in exploratory analysis.

3.5 Melting the Dataset

An ID column was added as df[’ID’] = df.index to uniquely identify each row. The dataset was then melted using pd.melt to transform it from wide to long format:

- id$_vars = ['ID',' variety'] : Keptasidentifiercolumns.$

The melted dataset has 600 rows (150 flowers × 4 measurements per flower) and 4 columns: ID, variety, measurement, and value. This format is useful for statistical analysis or visualization, as it groups all measurements into a single column.

3.6 Pivoting to Wide Format

The melted dataset was pivoted back to a wide format using melted$_{df.pivot}:$

- index=[’variety’, ’ID’]: Rows are indexed by variety and ID.

  columns=’measurement’: Measurement types become new columns.

  values=’value’: Values fill the new columns.

The reset$_index()functionwasappliedtomake$variety$andIDregularcolumns.Theresultingdataseth$

4 Results and Discussion

The analysis demonstrates key pandas operations:

- **Filtering**: Isolated specific species for focused analysis.
- **Merging**: Combined subsets to study multiple species together.
- **Sorting**: Revealed patterns, like Virginica's larger petal lengths.
- **Transposing**: Changed the dataset's perspective, though less practical here.
- **Melting**: Converted to a long format, ideal for certain analyses.
- **Pivoting**: Restored the wide format, showing reversibility of transformations.

These operations highlight pandas' flexibility in reshaping data for different purposes, such as visualization, statistical modeling, or reporting. The Iris dataset's simplicity made it an ideal candidate for practicing these techniques.

5 Conclusion

This project successfully applied pandas to manipulate the Iris dataset, demonstrating filtering, merging, sorting, transposing, melting, and pivoting. Each operation served a specific purpose, from exploring species-specific traits to reshaping data for analysis. The analysis provides a solid foundation for understanding data manipulation in Python, applicable to more complex datasets in real-world scenarios.

For future work, the dataset could be used for machine learning tasks, such as classifying iris species, or for advanced visualizations to compare species measurements.

6 Presentation Tips for Oral Explanation

To deliver an effective oral presentation:

- **Introduce the Dataset**: Briefly describe the Iris dataset and its importance in data science.
- **Explain Each Step Simply**: For example, describe melting as "turning multiple measurement columns into one for easier analysis."
- **Use Visuals**: Show tables or diagrams of the dataset before and after operations like melting or pivoting.
- **Highlight Purpose**: Explain why each operation is useful (e.g., sorting to find trends, melting for plotting).

- **Engage the Audience**: Ask questions like, “What do you think we’d see if we sorted by sepal width instead?”
- **Be Concise**: Focus on one key takeaway per operation to avoid overwhelming the audience.
- **Practice Terminology**: Be comfortable with terms like “wide format” and “long format” to sound confident.
- **Conclude with Impact**: Summarize how these techniques apply to real-world data analysis and suggest next steps.