

EFM Fall 2014, Week 8: pandas

Jason Phang, Allan Zhang, Zachry Wang

November 19, 2014

Table of Contents

1 pandas

2 pandas Demonstration

3 Looking Ahead

Table of Contents

1 pandas

2 pandas Demonstration

3 Looking Ahead

Introduction to pandas

Pandas

Introduction to pandas

Pandas

- Pandas is the culmination of all the Python you've learnt this quarter

Pandas

- Pandas is the culmination of all the Python you've learnt this quarter
- Go-to toolset for data manipulation, visualization, summarization and analysis

Introduction to pandas

Pandas

- Pandas is the culmination of all the Python you've learnt this quarter
- Go-to toolset for data manipulation, visualization, summarization and analysis
- Built by Wes McKinney, a software engineer at AQR, to facilitate research

Introduction to pandas

Pandas

- Pandas is the culmination of all the Python you've learnt this quarter
- Go-to toolset for data manipulation, visualization, summarization and analysis
- Built by Wes McKinney, a software engineer at AQR, to facilitate research
- Incorporates NumPy, matplotlib, and some of SciPy

Import pandas

```
import pandas as pd
```


Introduction to pandas

Pandas

- Pandas is the culmination of all the Python you've learnt this quarter
- Go-to toolset for data manipulation, visualization, summarization and analysis
- Built by Wes McKinney, a software engineer at AQR, to facilitate research
- Incorporates NumPy, matplotlib, and some of SciPy

Import pandas

```
import pandas as pd
```

Series and Data Frames

Series and Data Frames

1. **Series** - A column of data (numbers, strings, times, etc)

Series and Data Frames

1. **Series** - A column of data (numbers, strings, times, etc)
 - Built on NumPy arrays

1. **Series** - A column of data (numbers, strings, times, etc)
 - Built on NumPy arrays
 - Comes with an Index (also another NumPy Array). Index may be non-numerical

1. **Series** - A column of data (numbers, strings, times, etc)
 - Built on NumPy arrays
 - Comes with an Index (also another NumPy Array). Index may be non-numerical
 - Example: Price of a stock over time

1. **Series** - A column of data (numbers, strings, times, etc)
 - Built on NumPy arrays
 - Comes with an Index (also another NumPy Array). Index may be non-numerical
 - Example: Price of a stock over time
 - Example: Heights of people in group

Series and Data Frames

2. **Data Frame** - A table of data

2. **Data Frame** - A table of data

- Basically, a dictionary of Series, with a *lot* of additional features

2. **Data Frame** - A table of data

- Basically, a dictionary of Series, with a *lot* of additional features
- Based on R's data frames.

2. **Data Frame** - A table of data

- Basically, a dictionary of Series, with a *lot* of additional features
- Based on R's data frames.
- All columns share a single index

2. **Data Frame** - A table of data

- Basically, a dictionary of Series, with a *lot* of additional features
- Based on R's data frames.
- All columns share a single index
- Example: Price of multiple stocks over time

2. **Data Frame** - A table of data

- Basically, a dictionary of Series, with a *lot* of additional features
- Based on R's data frames.
- All columns share a single index
- Example: Price of multiple stocks over time
- Example: Heights, Weights, Age, Race of people in group

Table of Contents

1 pandas

2 pandas Demonstration

3 Looking Ahead

Series Features

Covered in demonstration (See IPython Notebook):

Series Features

Covered in demonstration (See IPython Notebook):

- Initialization of Series

Series Features

Covered in demonstration (See IPython Notebook):

- Initialization of Series
- Accessing/Modifying a Series index, indexing

Covered in demonstration (See IPython Notebook):

- Initialization of Series
- Accessing/Modifying a Series index, indexing
- Plotting

Covered in demonstration (See IPython Notebook):

- Initialization of Series
- Accessing/Modifying a Series index, indexing
- Plotting
- NumPy-style manipulation

Covered in demonstration (See IPython Notebook):

- Initialization of Series
- Accessing/Modifying a Series index, indexing
- Plotting
- NumPy-style manipulation
- Convenience Methods (e.g. max, diff, valid, sort)

Covered in demonstration (See IPython Notebook):

- Initialization of Series
- Accessing/Modifying a Series index, indexing
- Plotting
- NumPy-style manipulation
- Convenience Methods (e.g. max, diff, valid, sort)
- Apply

Data Frame Features

Covered in demonstration (See IPython Notebook):

Data Frame Features

Covered in demonstration (See IPython Notebook):

- Initialization, reading from CSV

Data Frame Features

Covered in demonstration (See IPython Notebook):

- Initialization, reading from CSV
- Accessing/Modifying columns

Data Frame Features

Covered in demonstration (See IPython Notebook):

- Initialization, reading from CSV
- Accessing/Modifying columns
- Convenience Methods

Data Frame Features

Covered in demonstration (See IPython Notebook):

- Initialization, reading from CSV
- Accessing/Modifying columns
- Convenience Methods
- **Adding rows is extremely inefficient**

Additional Features

Covered in demonstration (See IPython Notebook):

Additional Features

Covered in demonstration (See IPython Notebook):

- Moving Averages

Additional Features

Covered in demonstration (See IPython Notebook):

- Moving Averages
- Built-in APIs

Additional Features

Covered in demonstration (See IPython Notebook):

- Moving Averages
- Built-in APIs
- Regressions in **pandas**

pandas Summary

- Almost every kind of data analysis you encounter can be done through pandas

- Almost every kind of data analysis you encounter can be done through pandas
- **pandas is life, pandas is love**

Table of Contents

1 pandas

2 pandas Demonstration

3 Looking Ahead

Looking Ahead

What we've covered this quarter

Looking Ahead

What we've covered this quarter

- Python basics

Looking Ahead

What we've covered this quarter

- Python basics
- Libraries: NumPy, pandas

Looking Ahead

What we've covered this quarter

- Python basics
- Libraries: NumPy, pandas
- Modern Portfolio Theory, CAPM

Where do we go from here?

You know just enough to be dangerous

You know just enough to be dangerous

- Exploit publicly available data (FRED, Quandl, Google/Yahoo! Finance, web-scraping, etc)

You know just enough to be dangerous

- Exploit publicly available data (FRED, Quandl, Google/Yahoo! Finance, web-scraping, etc)
- Think actively about how to test hypotheses, simulate things, extend what you learn in class, etc

You know just enough to be dangerous

- Exploit publicly available data (FRED, Quandl, Google/Yahoo! Finance, web-scraping, etc)
- Think actively about how to test hypotheses, simulate things, extend what you learn in class, etc
- Start reading papers! (NBER, JoF, all available through the UChicago network)

You know just enough to be dangerous

- Exploit publicly available data (FRED, Quandl, Google/Yahoo! Finance, web-scraping, etc)
- Think actively about how to test hypotheses, simulate things, extend what you learn in class, etc
- Start reading papers! (NBER, JoF, all available through the UChicago network)

Keep learning

You know just enough to be dangerous

- Exploit publicly available data (FRED, Quandl, Google/Yahoo! Finance, web-scraping, etc)
- Think actively about how to test hypotheses, simulate things, extend what you learn in class, etc
- Start reading papers! (NBER, JoF, all available through the UChicago network)

Keep learning and have a great winter break!