



Computer
Science

CSC196: Analyzing Data

Course Wrapup

Jason Pacheco and Cesim Erten

Outline

- Pandas Overview
- Course Recap
- Additional Resources

Outline

- **Pandas Overview**
- Course Recap
- Additional Resources

Pandas



Open source library for data handling and manipulation in high-performance environments.

Installation If you are using Anaconda package manager,

```
conda install pandas
```

Or if you are using PyPi (pip) package manager,

```
pip install pandas
```

See Pandas documentation for more detailed instructions
https://pandas.pydata.org/docs/getting_started/install.html

DataFrame

Primary data structure : Essentially a table

The diagram illustrates a DataFrame as a table with columns and rows. The columns are labeled 'Name', 'Team', 'Number', 'Position', and 'Age'. The rows are indexed from 0 to 6. Annotations include 'Columns' pointing to the column headers, 'Rows' pointing to the row indices, and 'Data' pointing to the data cells. A small logo is visible in the bottom right corner of the diagram.

| | <i>Name</i> | <i>Team</i> | <i>Number</i> | <i>Position</i> | <i>Age</i> |
|---|-----------------|----------------|---------------|-----------------|------------|
| 0 | Avery Bradley | Boston Celtics | 0.0 | PG | 25.0 |
| 1 | John Holland | Boston Celtics | 30.0 | SG | 27.0 |
| 2 | Jonas Jerebko | Boston Celtics | 8.0 | PF | 29.0 |
| 3 | Jordan Mickey | Boston Celtics | NaN | PF | 21.0 |
| 4 | Terry Rozier | Boston Celtics | 12.0 | PG | 22.0 |
| 5 | Jared Sullinger | Boston Celtics | 7.0 | C | NaN |
| 6 | Evan Turner | Boston Celtics | 11.0 | SG | 27.0 |

DataFrame Example

Create and print an entire DataFrame

```
# import pandas as pd
import pandas as pd

# list of strings
lst = ['Geeks', 'For', 'Geeks', 'is',
       'portal', 'for', 'Geeks']

# Calling DataFrame constructor on list
df = pd.DataFrame(lst)
print(df)
```

| | 0 |
|---|--------|
| 0 | Geeks |
| 1 | For |
| 2 | Geeks |
| 3 | is |
| 4 | portal |
| 5 | for |
| 6 | Geeks |

DataFrame Example

Can create named columns using dictionary

```
import pandas as pd

# initialise data of lists.
data = {'Name':['Tom', 'nick', 'krish', 'jack'],
        'Age':[20, 21, 19, 18]}

# Create DataFrame
df = pd.DataFrame(data)

# Print the output.
print(df)
```

| | Name | Age |
|---|-------|-----|
| 0 | Tom | 20 |
| 1 | nick | 21 |
| 2 | krish | 19 |
| 3 | jack | 18 |

DataFrame : Selecting Columns

Select columns to print by name,

```
# Import pandas package
import pandas as pd

# Define a dictionary containing employee data
data = {'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
        'Age':[27, 24, 22, 32],
        'Address':['Delhi', 'Kanpur', 'Allahabad', 'Kannauj'],
        'Qualification':['Msc', 'MA', 'MCA', 'Phd']}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data)

# select two columns
print(df[['Name', 'Qualification']])
```

| | Name | Qualification |
|---|--------|---------------|
| 0 | Jai | Msc |
| 1 | Princi | MA |
| 2 | Gaurav | MCA |
| 3 | Anuj | Phd |

DataFrame : Selecting Rows

Select columns to print by name,

```
import pandas as pd
import numpy as np

# Define a dictionary containing employee data
data = {'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
        'Age':[27, 24, 22, 32],
        'Address':['Delhi', 'Kanpur', 'Allahabad', 'Kannauj'],
        'Qualification':['Msc', 'MA', 'MCA', 'Phd']}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data)

# Print rows 1 & 2
row = df.loc[1:2]
print(row)
```

Output

| | Name | Age | Address | Qualification |
|---|--------|-----|-----------|---------------|
| 1 | Princi | 24 | Kanpur | MA |
| 2 | Gaurav | 22 | Allahabad | MCA |

DataFrame : Selecting Rows

`head()` and `tail()` select rows from beginning / end

```
import pandas as pd
import numpy as np

# Define a dictionary containing employee data
data = {'Name': ['Jai', 'Princi', 'Gaurav', 'Anuj'],
        'Age': [27, 24, 22, 32],
        'Address': ['Delhi', 'Kanpur', 'Allahabad', 'Kannauj'],
        'Qualification': ['Msc', 'MA', 'MCA', 'Phd']}

# Convert the dictionary into DataFrame
df = pd.DataFrame(data)

# Print first / last rows
first2 = df.head(2)
last2 = df.tail(2)
print(first2)
print('\n', last2)
```

Output

| | Name | Age | Address | Qualification |
|---|--------|-----|-----------|---------------|
| 0 | Jai | 27 | Delhi | Msc |
| 1 | Princi | 24 | Kanpur | MA |
| | Name | Age | Address | Qualification |
| 2 | Gaurav | 22 | Allahabad | MCA |
| 3 | Anuj | 32 | Kannauj | Phd |

Reading Data from Files

Easy reading / writing of standard formats,

Output

```
df = pd.read_json("data.json")
print(df)
df.to_csv("data.csv", index=False)
df_csv = pd.read_csv("data.csv")
print(df_csv.head(2))
```

| | Duration | Pulse | Maxpulse | Calories |
|-----|----------|-------|----------|----------|
| 0 | 60 | 110 | 130 | 409.1 |
| 1 | 60 | 117 | 145 | 479.0 |
| 2 | 60 | 103 | 135 | 340.0 |
| 3 | 45 | 109 | 175 | 282.4 |
| 4 | 45 | 117 | 148 | 406.0 |
| .. | ... | ... | ... | ... |
| 164 | 60 | 105 | 140 | 290.8 |
| 165 | 60 | 110 | 145 | 300.4 |
| 166 | 60 | 115 | 145 | 310.2 |
| 167 | 75 | 120 | 150 | 320.4 |
| 168 | 75 | 125 | 150 | 330.4 |

[169 rows x 4 columns]

| | Duration | Pulse | Maxpulse | Calories |
|---|----------|-------|----------|----------|
| 0 | 60 | 110 | 130 | 409.1 |
| 1 | 60 | 117 | 145 | 479.0 |

Data Structure Conversions

Working with DataFrames outside of Pandas can be tricky,

```
df['Duration']
```

We can easily convert to built-in types, for example to a list (e.g. to use in Numpy or whatever),

```
0      60
1      60
2      60
3      45
4      45
..
164    60
165    60
166    60
167    75
168    75
Name: Duration, Length: 169, dtype: int64
```

```
L = df['Duration'].to_list()
print(L)
```

```
[60, 60, 60, 45, 45, 60, 60, 45, 30, 60, 60, 60, 60, 60, 60, 60, 60, 60, 45, 60, 45, 60, 45, 60, 45, 60, 60, 60, 60, 60, 60, 60, 45, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 45, 45, 60, 60, 80, 60, 60, 30, 60, 60, 45, 20, 45, 210, 160, 160, 45, 20, 180, 150, 150, 20, 300, 150, 60, 90, 150, 45, 90, 45, 45, 120, 270, 30, 45, 30, 120, 45, 30, 45, 120, 45, 20, 180, 45, 30, 15, 20, 20, 30, 25, 30, 90, 20, 90, 90, 90, 30, 30, 180, 30, 90, 210, 60, 45, 15, 45, 60, 60, 60, 60, 60, 60, 30, 45, 60, 60, 60, 60, 60, 60, 60, 90, 60, 60, 60, 60, 60, 60, 20, 45, 45, 45, 20, 60, 60, 45, 45, 60, 45, 60, 60, 30, 60, 60, 60, 60, 60, 30, 60, 60, 60, 60, 60, 60, 60, 60, 60, 60, 30, 30, 45, 45, 45, 60, 60, 60, 75, 75]
```

Summary Statistics


Easily compute summary statistics on data

```
print('Min: ', df['Duration'].min())  
print('Max: ', df['Duration'].max())  
print('Median: ', df['Duration'].median())
```

```
Min: 15  
Max: 300  
Median: 60.0
```

Can also count occurrences of
unique values,

```
df['Duration'].value_counts()
```



```
60    79  
45    35  
30    16  
20     9  
90     8  
150    4  
120    3  
180    3  
15     2  
75     2  
160    2  
210    2  
270    1  
25     1  
300    1  
80     1  
Name: Duration, dtype: int64
```

Outline

- Pandas Overview
- **Course Recap**
- Additional Resources

Course Overview

Course Objective *develop a solid fundamental understanding of probability and statistics and learn how to apply them to a diverse set of problems.*

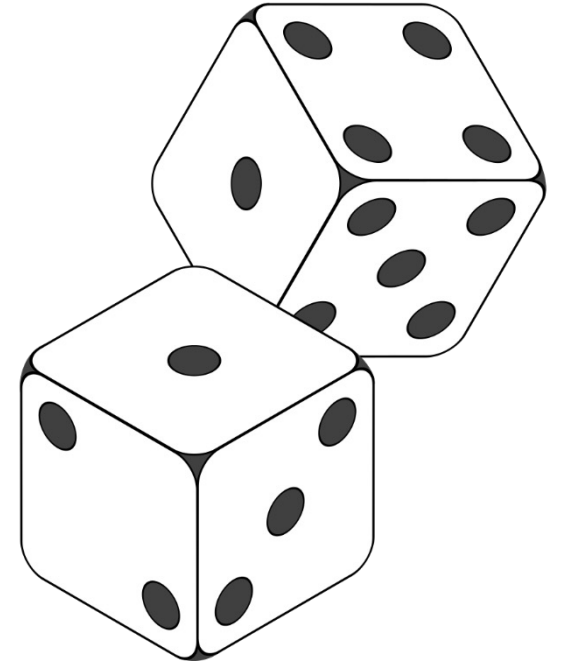
| Probability | Statistical Estimation | Hypothesis Testing | Bayesian Probability |
|---|---|--------------------------------|--|
| Random events / variables, distributions / densities, moments | Sampling distributions, central limit, one- and two-sample estimation | Confidence intervals, p-values | Bayes' rule, prior / posterior, credible intervals, model validation |

Probability and Statistics

Suppose we roll two fair dice...

- What are the possible outcomes?
- What is the *probability* of rolling **even** numbers?

... this is an **experiment** or **random process**.



We learned how to...

- Mathematically formulate outcomes and their probabilities?
- Describe characteristics of random processes
- Estimate unknown quantities (e.g. are the dice actually fair?)
- Characterize the uncertainty in random outcomes
- Identify and measure dependence among random quantities

Statistical Estimation

Sampling



Target Population



Sample

Use Sample to
Estimate Population

Estimation

Population Mean

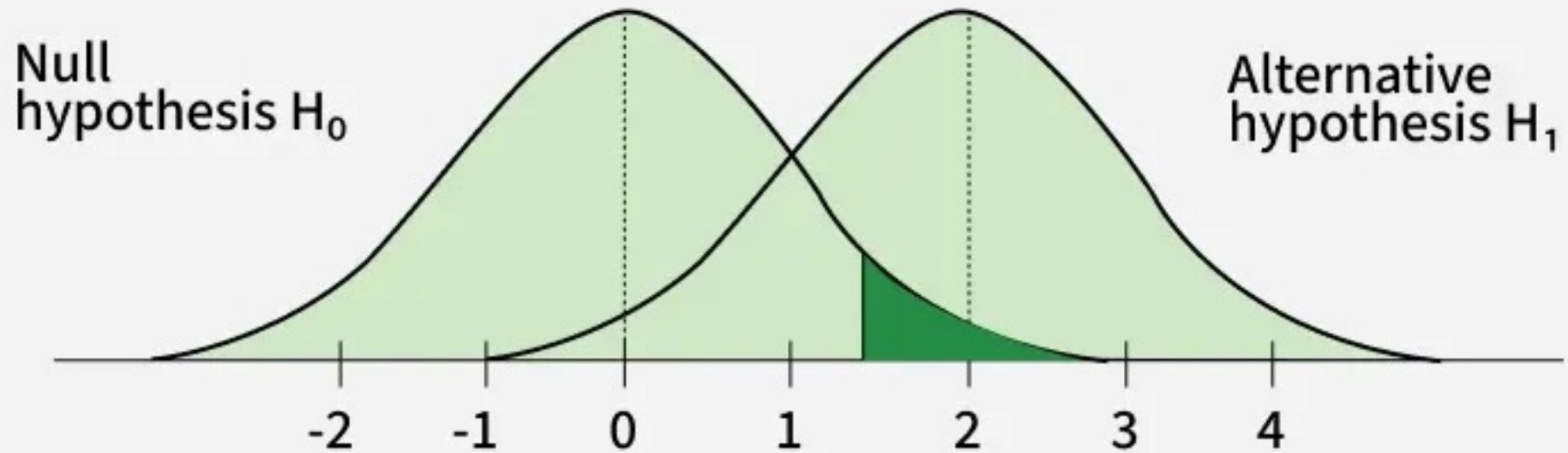
$$\mu = \frac{\sum_{i=1}^N x_i}{N}$$

Sample Mean

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{n}$$

Hypothesis Testing

Hypothesis Testing



Bayesian Probability

Posterior represents all uncertainty after observing data...

prior probability

likelihood function for the parameters

$$p(\theta | y) = \frac{p(\theta)p(y | \theta)}{p(y)}$$

posterior probability

Marginal likelihood
or: normalizer

The diagram illustrates the Bayesian formula with four arrows pointing from descriptive text to the corresponding terms in the equation. An arrow points from 'prior probability' to $p(\theta)$. Another arrow points from 'likelihood function for the parameters' to $p(y | \theta)$. A third arrow points from 'posterior probability' to $p(\theta | y)$. The fourth arrow points from 'Marginal likelihood or: normalizer' to $p(y)$.

Outline

- Pandas Overview
- Course Recap
- **Additional Resources**

Additional Relevant Courses

- CSC 280 : Introduction to AI
- CSC 380 : Principles of Data Science
- CSC 480 : Principles of Machine Learning
- CSC 444 : Introduction to Data Visualization

Videos

[3Blue1Brown](#)

- Accessible videos on a variety of math topics
- Nicely produced, engaging graphics
- A number of ML / Data Science / Statistics topics covered



Steve Brunton – [YouTube Channel](#)

- More detailed videos on math / engineering topics
- Good linear algebra and machine learning videos
- Associated book,

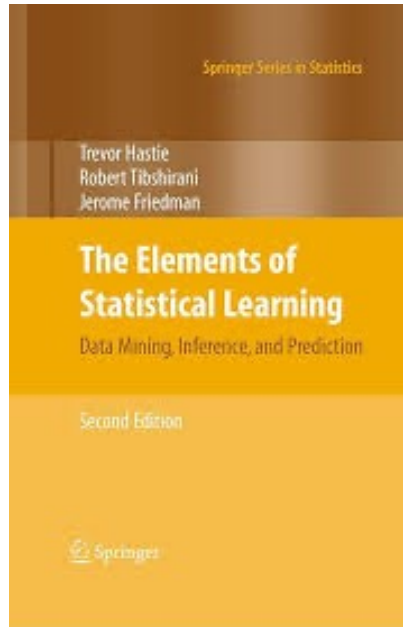
[Data-Driven Science and Engineering : ML, Dynamical Systems, and control](#)

Videos

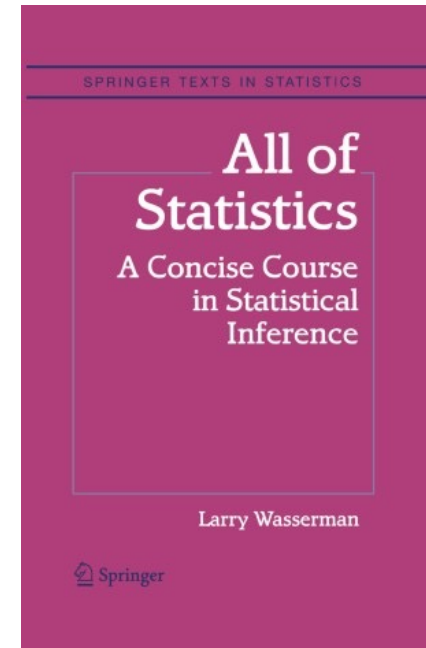
[MIT Open Courseware](#)

- Lots of topics freely available
- Excellent Linear Algebra course by Prof. Gilbert Strang ([YouTube lectures](#))
- All assignments and exams available online

Textbooks



Hastie et al., "The Elements of Statistical Learning 2nd Ed."
Springer, 2009
([UA Library](#))



Wasserman, L. "All of Statistics." Springer, 2004
([Springer](#))

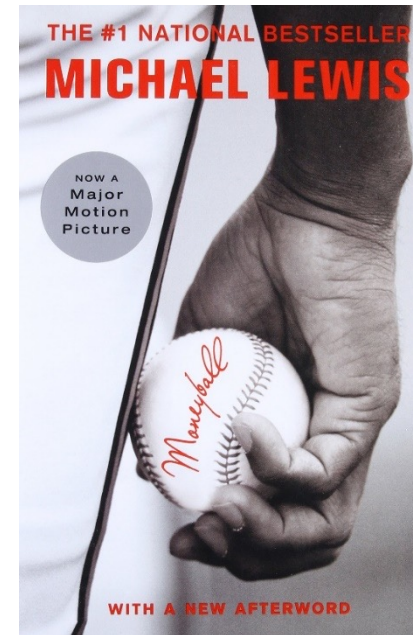
Non-Textbooks

new york times bestseller
noise and the noi
the signal and th
and the noise and
the noise and the
why so many noi
predictions fail—
but some don't th
and the noise and
nate silver the no

"Could turn out to be one of the more momentous books of the decade." —*The New York Times Book Review*



Silver, N. "The Signal and The Noise."
Penguin, 2015



Lewis, M. "Moneyball." W. W. Norton, 2011