Contents

Week 1 Lecture Notes: Introduction & Descriptive Statistics

---

1. Welcome and Course Overview

Welcome to **Quantitative Methods for Social Sciences**!
In this course, you will learn to apply powerful computational tools—**Python** and **R**—to perform statistical analysis tailored for social science research.

Our focus will be on real-world data analysis skills, covering descriptive and inferential statistics, regression models, Bayesian methods, and causal inference.

---

2. Introduction to Python Basics

Python is a versatile programming language widely used for data analysis.
You will interact with Python mostly via **Jupyter notebooks**, where code and explanatory text live together.

Python Data Types and Variables

```
### Numbers

age = 30
height = 1.75  # meters

### Strings

name = "Alice"

### Boolean

is_student = True
```

```
### List (ordered collection)

scores = [85, 90, 78]

print(f"{name} is {age} years old.")

## Jupyter Notebook Basics

    Code cells: Run Python code

    Markdown cells: Write formatted text

    Run a cell with Shift + Enter

---

## 3. Introduction to R Basics

R is a specialized language for statistics and data visualization,
with rich libraries for social science applications.

### Basic R Syntax

```r
# Numeric vector
ages <- c(23, 35, 42)

# Character vector
names <- c("Alice", "Bob", "Charlie")

# Logical vector
is_student <- c(TRUE, FALSE, TRUE)

print(paste("The first person is", names[1], "and is", ages[1], "years old."))
```

R in Jupyter

You can also run R inside Jupyter notebooks using the IRKernel.

---

4. Descriptive Statistics

Descriptive statistics summarize and describe key features of data.

Measures of Central Tendency

- **Mean** (average)
- **Median** (middle value)
- **Mode** (most frequent value)

Measures of Variability

- **Variance**: average squared deviation from the mean
```

- **Standard Deviation**: square root of variance (in original units)
- **Interquartile Range (IQR)**: range between 25th and 75th percentile

Python Example (using `pandas`)

```python
import pandas as pd

data = pd.Series([5, 7, 8, 5, 9, 10, 5, 6])

print("Mean:", data.mean())
print("Median:", data.median())
print("Standard Deviation:", data.std())
```

R Example

```r
data <- c(5, 7, 8, 5, 9, 10, 5, 6)

mean(data)
median(data)
sd(data)
```

---

5. Data Loading and Exploration

Understanding your dataset is crucial before analysis.

Loading Data

- Python (`pandas`):

```python
import pandas as pd

df = pd.read_csv('data/sample_data.csv')
print(df.head())
```

- R:

```r
df <- read.csv('data/sample_data.csv')y
head(df)
```

Exploring Data

- Check data types, missing values
- Summarize variables

Python:

```python
df.info()
df.describe()
```

R:

```r
str(df)
summary(df)
```

---

6. Visualizing Data

Basic plots help understand distribution and spot outliers.

Python (using `matplotlib` and `seaborn`)

```python
import matplotlib.pyplot as plt
import seaborn as sns

sns.histplot(df['variable'])
plt.show()
```

R (using base plot or `ggplot2`)

```r
hist(df$variable)

# Or with ggplot2
library(ggplot2)
ggplot(df, aes(x = variable)) + geom_histogram()
```

---

7. Summary and Next Steps

- We covered core concepts in Python and R.
- Practiced descriptive statistics and basic data loading.
- Next week: dive deeper into data wrangling and visualization.

References and Further Reading

- *Python for Data Analysis*, Wes McKinney
- *R for Data Science*, Hadley Wickham & Garrett Grolemund
- *OpenIntro Statistics*, Diez et al. (free textbook)
- pandas documentation
- CRAN R Project

End of Week 1 Lecture Notes