



# برنامه نویسی با پایتون

سعید محققى

07: Working with Data

## ■ پروژه 7: آنالیز و پردازش داده در فایل csv (csv processing)

1. خواندن فایل csv (و xls و xlsx) و دسترسی به داده ها

2. استفاده از پکیج pandas

3. تغییر داده‌ها (تبدیل واحدهای قد و وزن)

4. محاسبه BMI هر فرد

5. ذخیره نتایج در فایل csv

### What will we learn? (Keywords)

- Read / Write CSV Files (+ xls, xlsx)
- Pandas Package
- Python Zip function
- List Comprehension

# خواندن فایل CSV

- روش 1 (خواندن به صورت فایل متنی)

```
with open("weather_data.csv") as data_file:  
    data = data_file.readlines()
```

نوع: List

هر خط در یک آیتم (شامل \n)

- روش 2 (استفاده از csv)

```
import csv  
with open("weather_data.csv") as data_file:  
    data = csv.reader(data_file)
```

نوع: \_csv.reader

هر خط در یک آیتم (بدون \n)

- روش 3 (استفاده از pandas)

```
import pandas  
data = pandas.read_csv("weather_data.csv")
```

نوع: DataFrame

دارای توابع متعدد برای پردازش داده

## عملیات بر روی DataFrame در Pandas

weather\_data.csv

day	temp	condition
Monday	12	Sunny
Tuesday	14	Rain
Wednesday	15	Rain
Thursday	14	Cloudy
Friday	21	Sunny
Saturday	22	Sunny
Sunday	24	Sunny

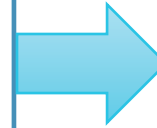
```
import pandas as pd
data = pd.read_csv("weather_data.csv")
data.head() #
data.tail()
data.to_dict()
data.temp # or data["temp"]
data.temp.to_list()
data.temp.mean()
data[data.day == "Monday"]
data[data.temp == data.temp.max()]
```

## ایجاد / ذخیره DataFrame در Pandas

```
data_dict = {"age": [30, 29],  
             "height": ['165', '170']}
```

```
df = pd.DataFrame(data_dict)
```

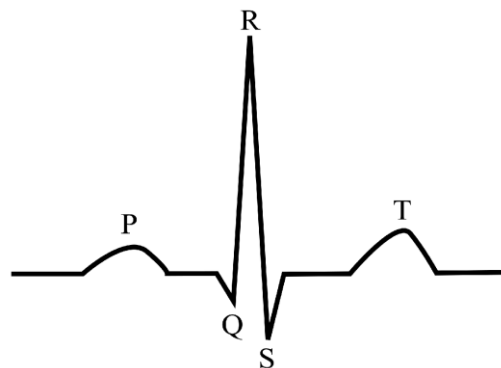
```
df = pd.DataFrame([(30, 165),  
                  (29, 170)],  
                  columns=('age', 'height'))
```



```
>>> df  
   age  height  
0   30     165  
1   29     170
```

```
>>> df.values  
array([[ 30, 165],  
       [ 29, 170]])
```

```
df.to_csv("my_data.csv")  
df.to_excel("my_data.xlsx")  
df.to_json("my_data.json")  
df.to_html("my_data.html")
```



## What will we learn? (Keywords)

- Read .mat Files
- Numpy / Scipy Package
- Matplotlib Package

## ■ پردازش سیگنال ECG (ecg processing)

1. خواندن فایل mat (Matlab) و دسترسی به داده ها

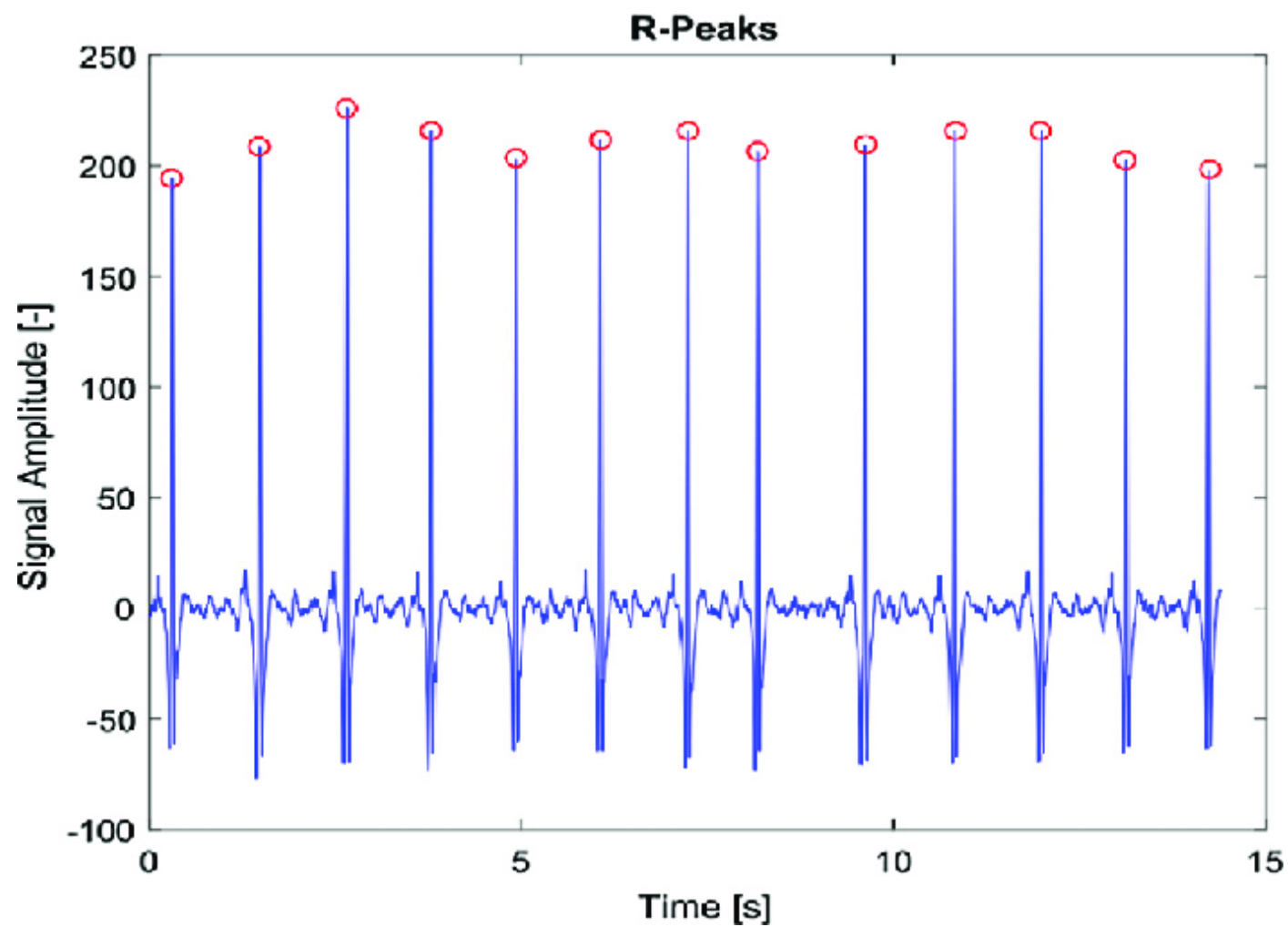
2. محاسبه پیک های سیگنال با Scipy

3. نمایش سیگنال

4. نمایش نقاط پیک روی سیگنال

<https://data.mendeley.com/datasets/7dybx7wyfn/3>

دانلود دیتا



■ پیک های R در سیگنال نوار قلب

## پکیج های Pandas و Matplotlib

<https://www.w3schools.com/python/pandas/default.asp>

آشنایی با Pandas



<https://pandas.pydata.org/>

[https://www.w3schools.com/python/matplotlib\\_intro.asp](https://www.w3schools.com/python/matplotlib_intro.asp)

آشنایی با Matplotlib



<https://matplotlib.org/>

<https://matplotlib.org/stable/gallery/>

مثال های Matplotlib



## پکیج های Numpy و Scipy

<https://www.w3schools.com/python/numpy/default.asp>

آشنایی با Numpy

**NumPy** 

<https://numpy.org/>



**SciPy**

<https://www.w3schools.com/python/scipy/index.php>

آشنایی با Scipy

<http://scipy.org/>

## ■ پروژه 8: پردازش تصویر (image processing)

1. خواندن تصاویر

2. تغییر اندازه و چرخش تصاویر

3. بهبود روشنایی، کنتراست و رنگ تصاویر

4. نمایش تصاویر

5. ذخیره تصاویر

### What will we learn? (Keywords)

- Read / Write images
- Image Processing with Pillow
- Show Images in Matplotlib

- سه پکیج اصلی برای کار با تصاویر



<https://docs.opencv.org/master/>  
[OpenCV Python Tutorials](#)



<https://pillow.readthedocs.io/>  
[Pillow Tutorials](#)



<https://matplotlib.org/>  
[Matplotlib Tutorials](#)

\_\_\_\_\_ OpenCV is 1.4 Times faster than PIL \_\_\_\_\_



[cv2 vs PIL](#)