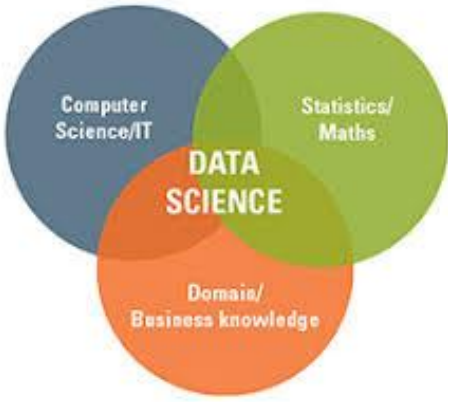
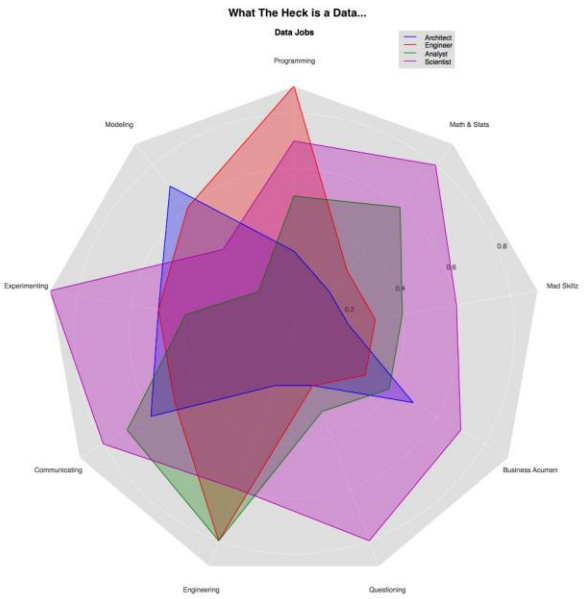
Programming for Data Science (with Python)

Le Trong Ngoc – http://fit.iuh.edu.vn/giangvien@letrongngoc

Contents

- Introduction to Data Science and Data Analysis
- Introduction to Python for Data Science
- Data Visualization with Python
- Statistical Thinking in Python
- Applied Machine Learning in Python





Data Analysis Process

- **Collecting Data** from various sources
- **Wrangling Data** to make it more reliable
- **Exploring Data** using statistics and visualizations
- **Transforming Data** to prepare it for modeling
- **Modeling Data** using the right machine learning algorithms
- **Evaluating** the results of the data models

→**interpret raw data by converting it into useful, and actionable intelligence**

Introduction to Python for Data Science

- Set up the Lab Environment
- Python basics
- List – A Data Structure
- Functions and Packages
- Numpy
- Plotting with Matplotlib
- Control Flow and Pandas

Set up the Lab Environment

- Python 3.5.x
- IDE: Spyder (*Scientific PYthon Development EnviRonment*)
- Python Library: NumPy, SciPy, SciKit-Learn, Pandas, and MatPlotLib
- **Anaconda / Spyder Installation**

Python Basics

- Case-sensitive name
- Example: caculate BMI

```
In [1]: height=1.69
In [2]: weight=76.5
In [3]:  bmi=weight/height**2
In [4]:  bmi
Out[4]:  26.784776443401846
```

- Type – float

```
In [6]: type(bmi)
Out[6]: float
```


Python Basics

- Type - int

```
In [7]: day_of_week = 5
In [8]: type(day_of_week)
Out[8]: int
```

- Type – string,text

```
In [10]: x="Introduction to Python for Data Science"
In [11]: y='Data Visualization with Python'
In [12]: type(y)
Out[12]: str
```

Python Basics

- Type – true,false
  In [**13**]: z= True
  In [**14**]: type(z)
  Out[**14**]: bool
- Different type = different behavior
  In [**15**]: 2 + 3
  Out[**15**]: 5
  In [**16**]: 'ab' + 'cd'
  Out[**16**]: 'abcd'

List – A Data Structure

- Python List
  In [**1**]: fam=[1.68, 1.56, 1.74, 1.34]
  In [**2**]: fam
  Out[**2**]: [1.68, 1.56, 1.74, 1.34]
  - ✓ Name a collection of values
  - ✓ Contain any type
  - ✓ Contain different types

List – A Data Structure

- Python List

```
In [3]: fam2 = [["liz", 1.73],
                ["emma", 1.68],
                ["mom", 1.71],
                ["dad", 1.89]]
In [4]: fam2
Out[4]: [['liz', 1.73], ['emma', 1.68], ['mom', 1.71], ['dad', 1.89]]
```


List – A Data Structure

- List Type

```
In [5]: type(fam)
Out[5]: list
In [6]: type(fam2)
Out[6]: list
```

  - ✓ Specific functionality
  - ✓ Specific behavior

List – A Data Structure

- Subsetting Lists

  In [1]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]

  In [2]: fam

  Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]

index:	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
	-8	**-7**	**-6**	**-5**	**-4**	**-3**	**-2**	**-1**

In [3]: fam[3]

Out[3]: 1.68

List – A Data Structure

- Subsetting Lists

  In [1]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]

  In [2]: fam

  Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]

index:	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
	-8	**-7**	**-6**	**-5**	**-4**	**-3**	**-2**	**-1**

In [4]: fam[-5]

Out[4]: 1.68

List – A Data Structure

- List Slicing

Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]

index: **0** **1** **2** **3** **4** **5** **6** **7**

In [**5**]: fam[3:5]

Out[**5**]: [1.68, 'mom']

[start : end]
inclusive exclusive

List – A Data Structure

- List Slicing

Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]

index: **0** **1** **2** **3** **4** **5** **6** **7**

In [**6**]: fam[:4]

Out[**6**]: ['liz', 1.73, 'emma', 1.68]

In [**7**]: fam[5:]

Out[**7**]: [1.71, 'dad', 1.89]

List – A Data Structure

- Changing list elements

```
In [1]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]
In [2]: fam
Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]

In [8]: fam[7] =1.68
In [9]: fam
Out[9]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.68]
```


List – A Data Structure

- Adding and removing elements

```
Out[9]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.68]

In [10]: fam=fam + ["me", 1.79]
In [11]: fam
Out[11]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.68, 'me', 1.79]

In [12]: del(fam[2])
In [13]: fam
Out[13]: ['liz', 1.73, 1.68, 'mom', 1.71, 'dad', 1.68, 'me', 1.79]
```

List – A Data Structure

- More

```
In [1]: x=["a","b","c"]
In [2]: y=x
In [3]: y
Out[3]: ['a', 'b', 'c']
In [4]: x[1]="z"
In [5]: x
Out[5]: ['a', 'z', 'c']
In [6]: y
Out[6]: ['a', 'z', 'c']
```


List – A Data Structure

- More

```
In [1]: x=["a","b","c"]
In [2]: y=list(x)
In [3]: z=x[:]

In [4]: x[1]="t"
In [5]: y
Out[5]: ['a', 'b', 'c']
In [6]: z
Out[6]: ['a', 'b', 'c']
```