上海立信会计金融学院
SHANGHAI LIXIN UNIVERSITY OF ACCOUNTING AND FINANCE

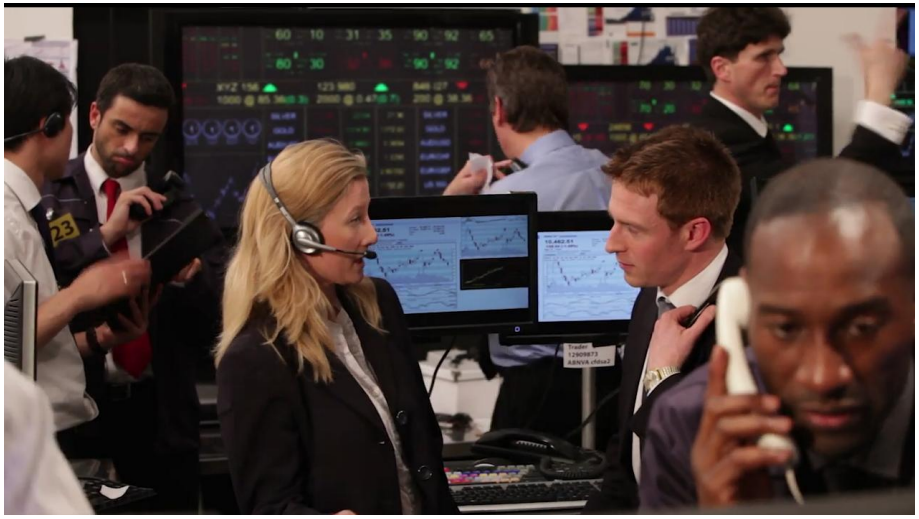
《Python金融数据分析》

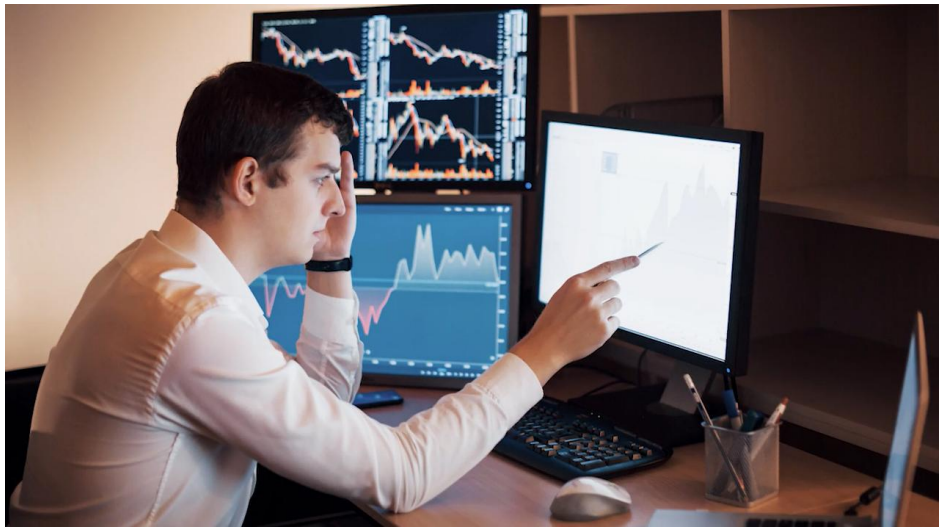
Hong Cheng（程宏）

School of Statistics and Mathematics

Shanghai LiXin University of Accounting and Finance

March 2022





资金委托给**股票经纪人**

股票经济人**提供最新的以及最必要的市场信息**给客户以方便做决策。

【**智能股票经纪人**】，可以自己进行股票交易。

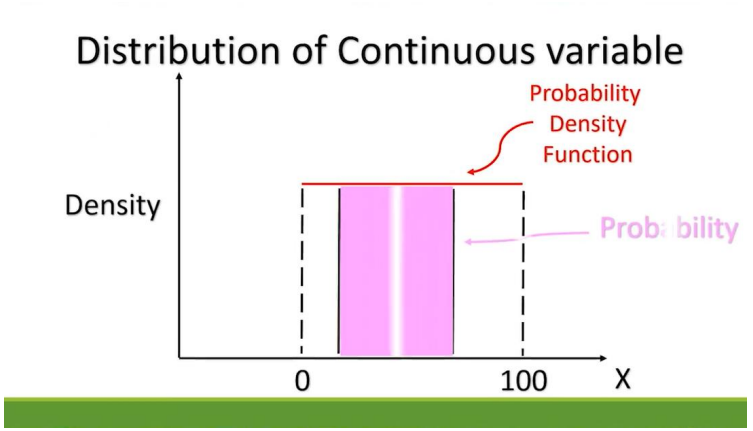
如何交易？？

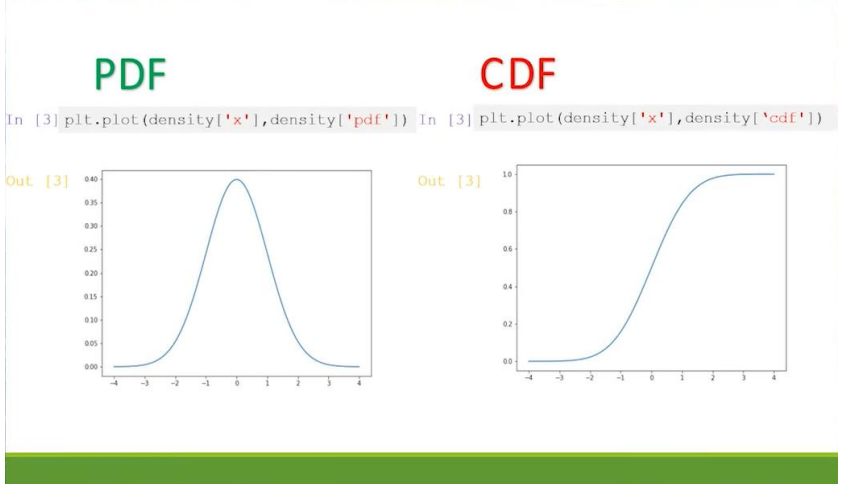
基于金融数据以及市场上的信息

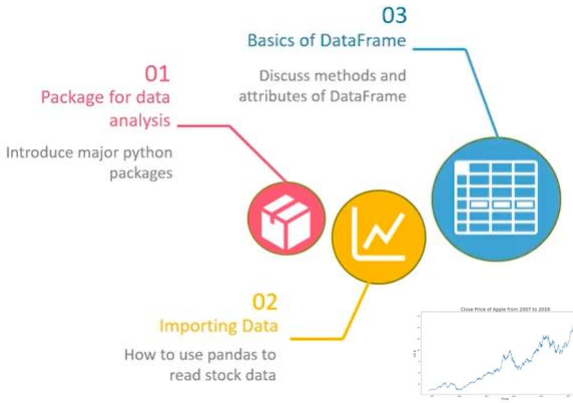
信息如何获取？？

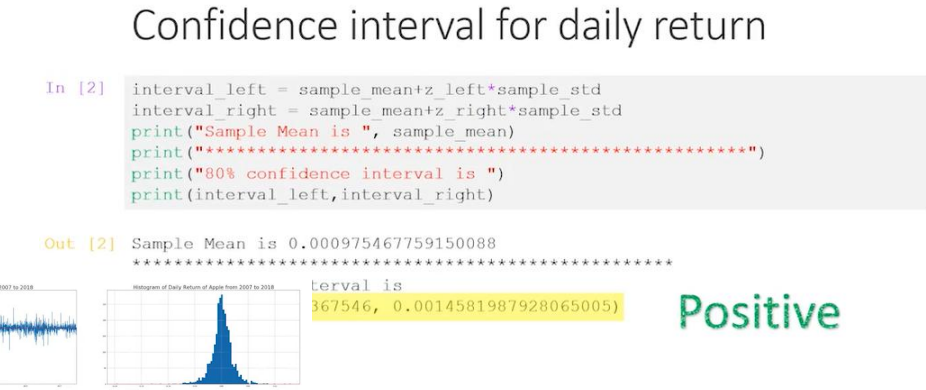
这些信息主要通过数据挖掘的方式从网络上抓取

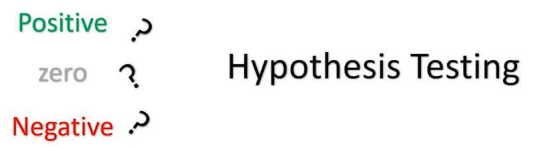
把Python和统计学概念进行组合
把它们应用到金融数据分析当中











运用统计学的概念 ⇒ 通过Python公式和语法组合来完成金融数据分析

学习如何**导入以及保存金融数据**，并且学习如何通过变量、数组和Python来利用已有的数据，使用**Python来实现并构建首个简单的股票交易模型**

Import pandas for DataFrame

```
In [1]  import pandas

In [2]  import pandas as pd
```

Methods of DataFrame

```
In [3]  fb.describe()
```

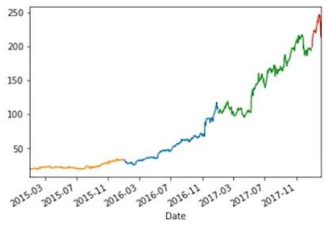
Out [3]

	Open	High	Low	Close	Adj Close	Volume
count	780.000000	780.000000	780.000000	780.000000	780.000000	7.800000e+02
mean	80.212705	81.285654	79.022397	80.264897	79.914215	1.204453e+07
std	64.226121	65.048907	63.190963	64.198375	64.327846	8.221848e+06
min	19.250000	19.500000	18.940001	19.139999	18.576082	1.311200e+06
25%	25.525000	26.085000	24.845000	25.475000	25.134513	7.215200e+06
50%	53.379999	54.034999	52.930000	53.420000	53.035403	9.728700e+06
75%	113.322502	115.779999	110.297499	113.702501	113.261238	1.408885e+07
max	245.770004	249.270004	244.449997	246.850006	246.850006	9.232320e+07

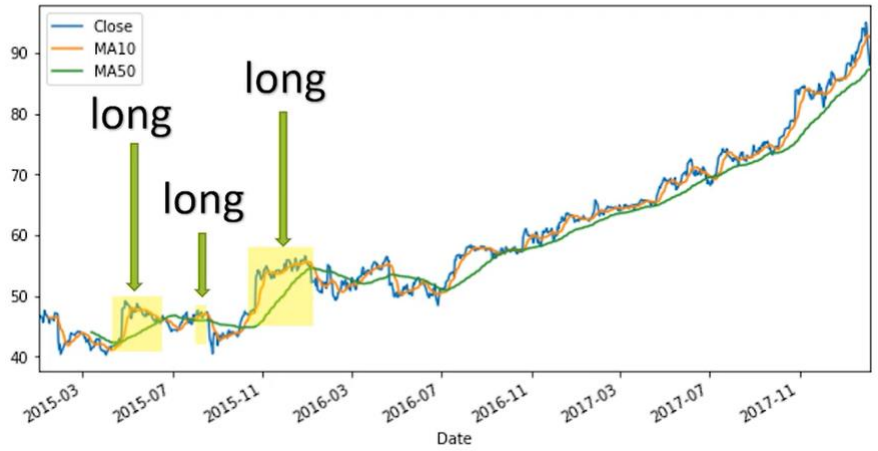
Visualizing stock price

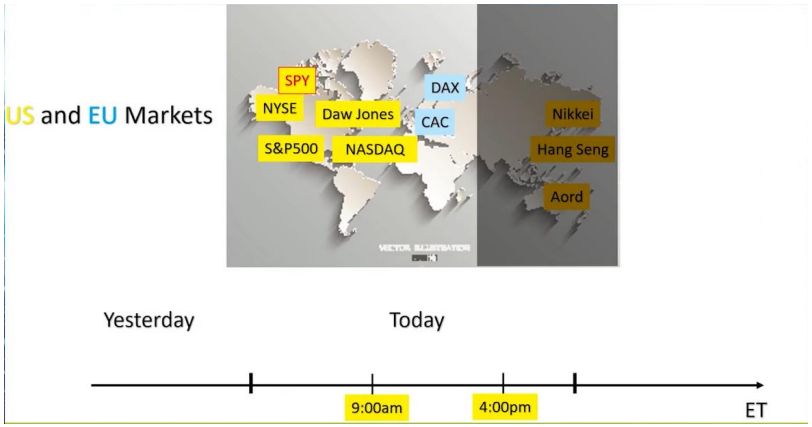
```
In [6]  fb.loc['2015-01-01':'2015-12-31','Close'].plot()# 2015
        fb.loc['2016-01-01':'2016-12-31','Close'].plot() # 2016
        fb.loc['2017-01-01':'2017-12-31','Close'].plot()# 2017
        fb.loc['2018-01-01':'2018-12-31','Close'].plot()# 2018
```

Out [6]



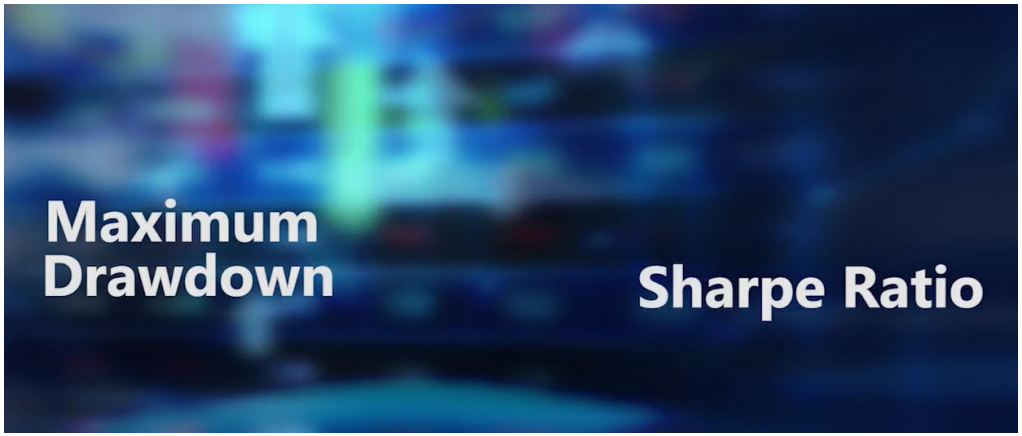
MA10 > MA50 “Long one share of stock”





在后面的课程涉及一个更高级的模型，即**使用线性回归模型来预测股票收益**

通过包括**统计标准**与**财务标准**等方法**评估模型的性能**，例如“夏普比率”和“最大跌幅”



jupyter Import data Last Checkpoint: 10/22/2018 (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Code

Import data

In this Jupyter Notebook, you will learn how to import data from CSV into Jupyter Notebook

In [3]:
```
#import the package "Pandas" into Jupyter Notebook
import pandas as pd
```

In [4]:
```
#We import the stock data of Facebook into Jupyter Notebook. The CSV file is located in
#We then name the DataFrame name as 'fb'
fb = pd.read_csv('../data/facebook.csv')
```

Instruction

Now is your turn to import the stock price of Microsoft (microsoft.csv), of which the CSV is located in tl

In [7]:
```
ms = pd.read_csv('../data/microsoft.csv')
```

In [8]:
```
# run this cell to ensure Microsoft's stock data is imported
print(ms.iloc[0, 0])
```

```
2014-12-31
```

jupyter Create new features and columns in DataFrame Last Checkpoint: 10/22/2018 (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

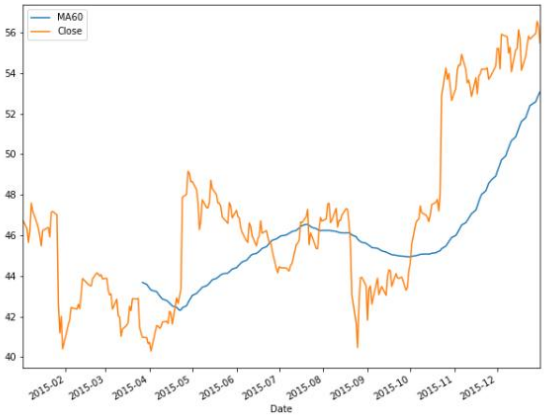
Code

In [ ]:
```
# You can use .rolling() to calculate any numbers of days' Moving Average. This is your turn to calculate "60 days"
# moving average of Microsoft, rename it as "ma60". And follow the codes above in plotting a graph

ms['ma60'] = None

#plot the moving average
plt.figure(figsize=(10, 8))
ms['ma60'].loc['2015-01-01':'2015-12-31'].plot(label='MA60')
ms['Close'].loc['2015-01-01':'2015-12-31'].plot(label='Close')
plt.legend()
plt.show()
```

Expected Output:

Jupyter notebook的编程环境

可以练习金融分析，以及课程案例



You may refer to the basic tutorial in https://www.datacamp.com/community/tutorials/tutorial-jupyter-notebook , which explains how to use Jupyter Notebooks.

For **learners** who **wish to install Jupyter Notebook locally in your computer**, you may also follow the same tutorial for installation instructions.

Jupyter Notebook Tutorial: The Definitive Guide

This tutorial explains how to install, run, and use Jupyter Notebooks for data science, including tips, best practices, and examples.

Nov 2019 · 25 min read

CONTENTS

SHARE

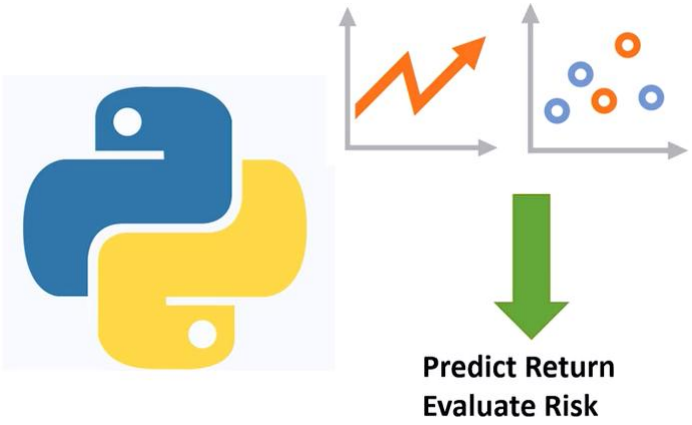
希望在本课程结束时你可以写pythony代码来构建统计模型从而帮助你进行金融分析。

最重要的是你可以通过模型评估建立的财务模型来确保其准确性与性能

Python的基础知识，其与金融数据的统计分析有关

首先需要了解Python在金融行业是怎样使用的

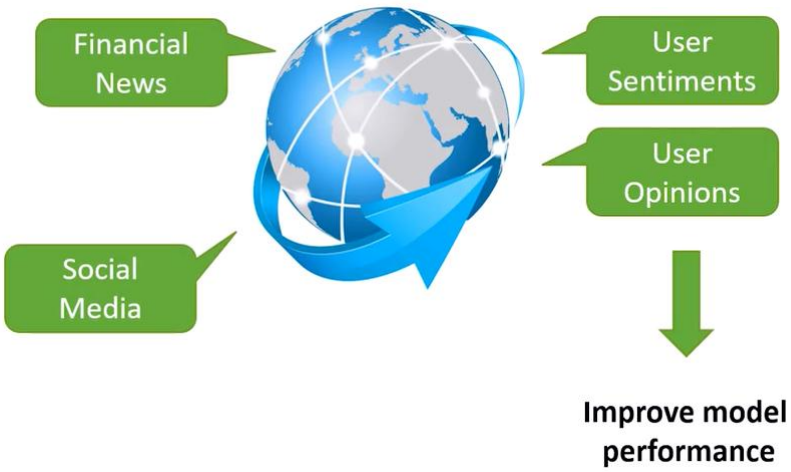
Build Models to predict returns and risks



投资银行的定量分析师与工程师**使用Python构建各种各样的模型**，从而**预测回报并评估风险**。

使用**Python爬取金融新闻，**来**挖掘**出**用户的想法**和**情感倾向**

Crawl data from the Internet



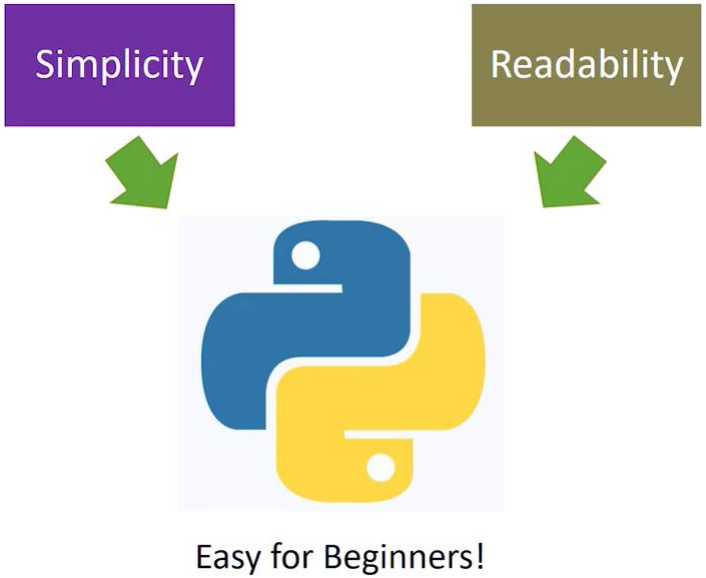
来自社交媒体的新数据来源可以极大地帮助定量分析师**改善模型的性能**。

Python不仅用于投资银行，它甚至在消费者银行中也被广泛使用。

Models in Consumer Banks

消费银行中的许多数据科学家都使用Python来**构建信用风险模型**。为降低贷款风险，他们可以**构建消费者行为模型**。来预测消费者的行为他们可以使用Python**构建推荐模型**在不同部门之间更准确地推荐新客户，这被称为"客户迁移" 。

Why is Python good for Financial Data Analysis?



简单性意味着Python的语法易于学习。

可读性意味着Python的代码易于理解。

Python成为主要编程语言的原因

- Python是具备动态语义、面向对象的解释型高级编程语言
- 可作为脚本或者“粘合剂”语言
- 简单、易学的语法强调可读性，可降低程序维护成本
- 支持模块和软件包，鼓励模块化和代码重用
- 解释程序和大量标准库可以免费取得、跨平台使用、随意分发

Python 语言的特征

- *开放源码*
- *解释型*
- *多重范型*
- *多用途*
- *跨平台*
- *动态类型*
- *缩进感知*
- *垃圾收集*

Python 简史

开发者：荷兰人吉多•范•罗苏姆（2018年7月让位）

1991年发行第一个版本Python 0.9.0

本课程使用Python 3.7

Python生态系统

» Python不仅是一种编程语言，也是一个生态系统

» 有大量的库与工具，可在需要时导入

» 自带的Python标准库（如标准的math库和其他可选库）

» Python交互式开发环境（最常见的是Ipython和Jupyter）

金融中的科技

- 科技成为了全球几乎所有金融机构的重要资产，具备导致竞争优势和劣势的潜力
- 银行和金融机构是每年在科技上投入最多的行业
- 科技发展是金融创业的业务引擎，对该行业的创新和效率增进有贡献，但也带来风险与监管的难度
- 充足的科技人才和合适的工具，是金融创新的关键
- 金融行业最受科技进步影响的，是金融交易决策与执行的速度及频率，只有现代科技才能有效应对
- 金融与数据分析越来越重要，面对的挑战主要是大数据和实时经济，需要更大规模的计算设施和更好的算法

用于金融的Python

金融和Python语法

- Python的优势
  - 语法与数学语法相近
  - 每条数学或算法语句都可以翻译为单行代码
  - 紧凑的向量化语法
  - 可以起到伪代码的作用，在高层次抽象和严格思想中求得平衡

从下面的例子中可以看到，Python实现金融算法简便易懂

示例：通过蒙特卡洛模拟方法估计欧式看涨期权的价值（BSM模型）

- 模型参数
  - 初始股票指数水平S_0=100；
  - 欧式看涨期权的行权价格K=105；
  - 到期时间T=1年；
  - 固定无风险短期利率r=5%；
  - 固定波动率 σ=20%。
- 公式1 BSM（1973）到期指数水平：
- 算法描述
  1. 从标准正态分布中取得I个（伪）随机数$z(i)$，$i \in \{1, 2, \cdots, I\}$。
  2. 为给定的$z(i)$和公式1-1计算所有到期指数水平$S_T(i)$。
  3. 计算到期时期权的所有内在价值$h_T(i)=\max(S_T(i)-K, 0)$。
  4. 通过公式1-2中给出的蒙特卡罗估算函数估计期权现值。
- 公式 2 **欧式期权的蒙特卡洛估算函数：**

$$C_0 \approx \mathrm{e}^{-rT} \frac{1}{I} \sum_I h_T(i)$$

示例：通过蒙特卡洛模拟方法估计欧式看涨期权的价值（续）

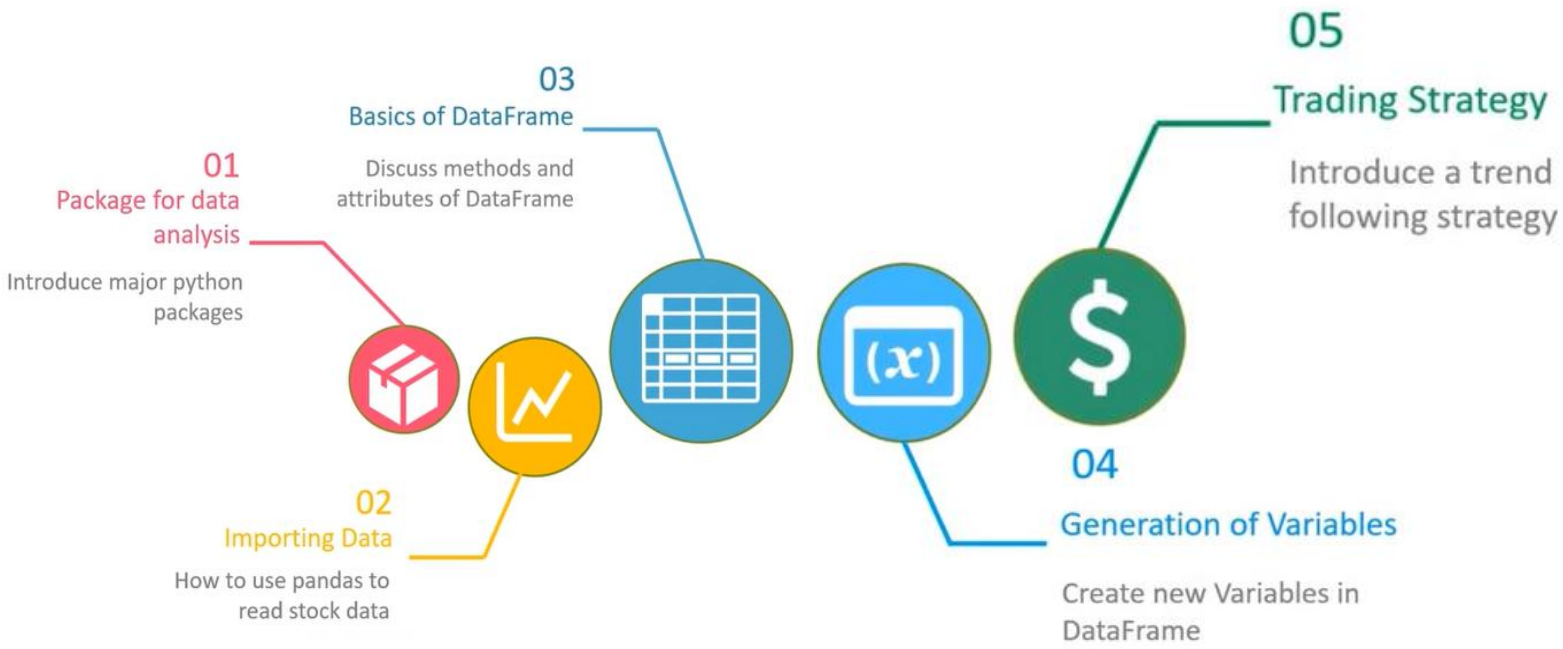
上述算法的Python翻译

```
In [6]: import math
        import numpy as np ❶

In [7]: S0 = 100. ❷
        K = 105. ❷
        T = 1.0 ❷
        r = 0.05 ❷
        sigma = 0.2 ❷

In [8]: I = 100000 ❷

In [9]: np.random.seed(1000) ❸

In [10]: z = np.random.standard_normal(I) ❹

In [11]: ST = S0 * np.exp((r - sigma ** 2 / 2) * T + sigma * math.sqrt(T) * z) ❺

In [12]: hT = np.maximum(ST - K, 0) ❻

In [13]: C0 = math.exp(-r * T) * np.mean(hT) ❼

In [14]: print('Value of the European call option: {:5.3f}.'.format(C0)) ❽
Value of the European call option: 8.019.
```

数据驱动和人工智能优先的金融学

- 最重要的一些金融理论缺乏实证，流行是因为符合人们的期望
- 科学方法始于数据，然后得出假设和理论再用数据测试，而计量金融学则与此相悖，原因是过去难以取得合适的数据
- 20世纪90年代初期起，金融机构所能得到的数据极具增加，可以专注于科学方法
- 数据是金融业的推动力，越来越多的服务产品向机构及个人提供海量数据，Python是重要的备选语言
- 取得大量金融数据后，应用人工智能（尤其是机器学习和深度学习）就更加容易、更有成果了
- AI重塑了金融行业，是一个令人兴奋的领域，Python则是AI的宠儿

通过以下步骤介绍Python教程



希望你能够在第一个Topic之后灵活运用股票数据
以实现你的各种奇妙想法。

01

Packages for data analysis

下面将介绍在本课程中用到的Python包

Pandas是一个python包，提供快速、灵活和富有表现力的数据结构。

它的**目标**是成为基本的高级构建模块，用于进行实际的现实世界数据分析。

Pandas名字衍生自术语 "panel data"（面板数据）和"Python data analysis"（Python数据分析）。

Pandas可以从各种文件格式比如CSV、JSON、SQL、Excel导入数据。

Pandas可以对各种数据进行运算操作，比如归并、再成形、选择，还有数据清洗和数据加工特征。

Pandas的主要数据结构是：
Series（一维数据）与DataFrame（二维数据）

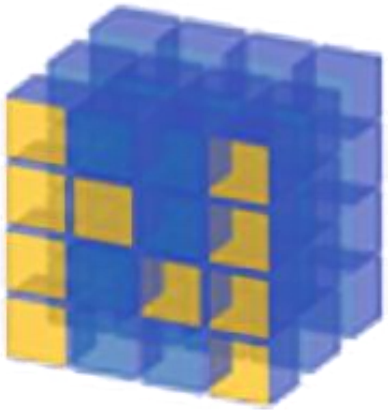
Excellent data structure for time series data
Pre-process data easily with DataFrame

Pandas

例如，Pandas中的**DataFrame**和**Series**是存储表和时间序列数据的优秀数据结构。

例如，使用DataFrame，我们可以轻松地预处理数据。
例如，**处理缺失值**，**计算成对相关性**。
在本课程中，我们将大量使用DataFrame。

Pandas一个强大的分析结构化数据的工具集，基础是**Numpy**（提供高性能的矩阵运算）。



NumPy(Numerical Python)是Python语言的一个扩展程序库，支持大量的维度数组与矩阵运算，此外也针对数组运算提供大量的**数学函数库**。

NumPy是一个运行速度非常快的数学库，主要用于数组计算，包含：

- 一个强大的N维数组对象ndarray
- 广播功能函数
- 整合C/C++/Fortran代码的工具
- 线性代数、傅里叶变换、随机数生成等功能

NumPy通常与**SciPy（Scientific Python）**和**Matplotlib（绘图库）**一起使用，这种组合广泛用于替代MatLab，是一个强大的科学计算环境，有助于我们通过Python学习数据科学或者机器学习。

SciPy是一个开源的Python算法库和数学工具包。

SciPy包含的模块有最优化、线性代数、积分、插值、特殊函数、快速傅里叶变换、信号处理和图像处理、常微分方程求解和其他科学与工程中常用的计算。

Matplotlib是Python的绘图库，它能让使用者很轻松地将数据图形化，并且提供多样化的输出格式。

Matplotlib可以用来绘制各种静态，动态，交互式的图表。

Matplotlib是一个非常强大的Python画图工具，可以使用该工具将很多数据通过图表的形式更直观的呈现出来。

Matplotlib可以绘制线图、散点图、等高线图、条形图、柱状图、3D 图形、甚至是图形动画等等。

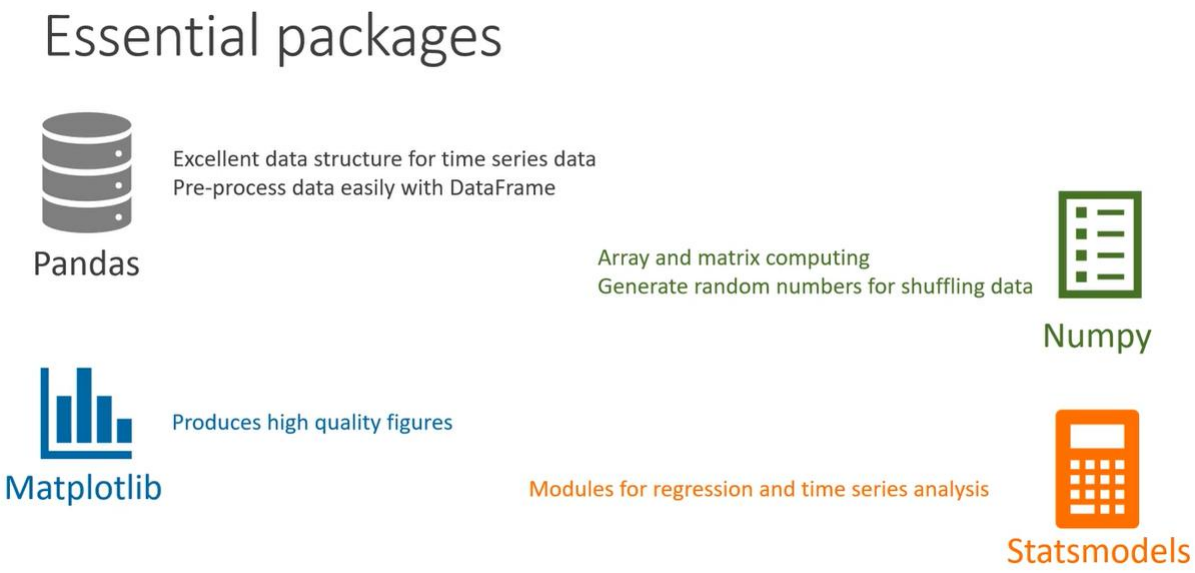
Statsmodels是一个Python库，用于拟合多种统计模型，执行统计测试以及数据探索和可视化。Statsmodels包含更多的“经典”频率学派统计方法，而贝叶斯方法和机器学习模型可在其他库中找到。

包含在statsmodels中的一些模型：

- 线性模型，广义线性模型和鲁棒线性模型
- 线性混合效应模型
- 方差分析（ANOVA）方法
- 时间序列过程和状态空间模型
- 广义的矩量法

Modules for regression and time series analysis

最常见的库有进行矩阵运算的**Numpy**、进行数据处理的**Pandas**、进行科学计算的**Scipy**、进行图形绘制及科学可视化的**Matplotlib**、用于拟合多种统计模型的**Statsmodels**、进行符号计算的**Sympy**以及方便进行机器学习任务的**Sklearn**。



Question

Pandas,Numpy,Matplotlib and Statsmodels are the 4 major packages that we are going to use in this course. You may want to get to know more about the software packages in the links below:

Pandas: https://pandas.pydata.org/about.html

Numpy: https://www.numpy.org/

Matplotlib: https://matplotlib.org/

Statsmodels: https://www.statsmodels.org/stable/index.html

Homework1： 使用Python中的Statsmodels评估线性模型

（1）生成随机数据

（2）根据随机数据生成线性模型

（3）通过OLS类拟合一个最小二乘线性回归

（4）使用模型的fit方法返回拟合的回归结果对象，该对象包含估计的模型参数和其他的诊断

（5）在results上调用summary方法可以打印出一个模型的诊断细节

Homework2： 使用Python中的Pandas完成以下任务：

（1）将DataFrame转换为NumPy数组，使用．values属性

（2）将数组再转换为DataFrame，可以传递一个含有列名的二维ndarray

（3）采用pandas的Categorical类型，产生一个非数字类型的列

采用的**数据如下**：

x0=[1,2,3,4,5],

x1=[0.01,-0.01,0.25,-4.1,0.],

y=[-1.5,0.,3.6,1.3,-2.]

非数字类型的列：

[a,b,a,a,b]

金融数据分析的第一步是获取数据

One of the popular format for storing data is CSV file

You will learn **how to import files with CSV format** and **how to save data into DataFrame structure**.

Import pandas for DataFrame

```
In [1]  import pandas

In [2]  import pandas as pd
```

pd.FunctionName

pd.ModuleName

Now, we want to **import two data files**, **facebook.csv** and microsoft.csv

These are historical stock data of Facebook and Microsoft. We can import these two files with two lines of codes like that. We are **using DataFrame to store data**, therefore we need a **pd**.

```
In [2]  fb=pd.DataFrame.from_csv('data/facebook.csv')
        ms=pd.DataFrame.from_csv('data/microsoft.csv')
```

The method **from_csv** help us to read the CSV files saved in the form of data

Pandas DataFrame

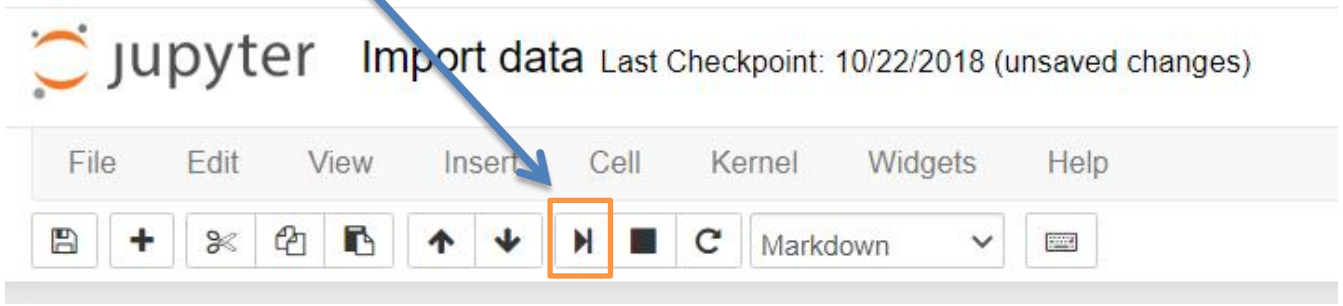
```
In [3]  print(type(fb))
        <class 'pandas.core.frame.DataFrame'>
```

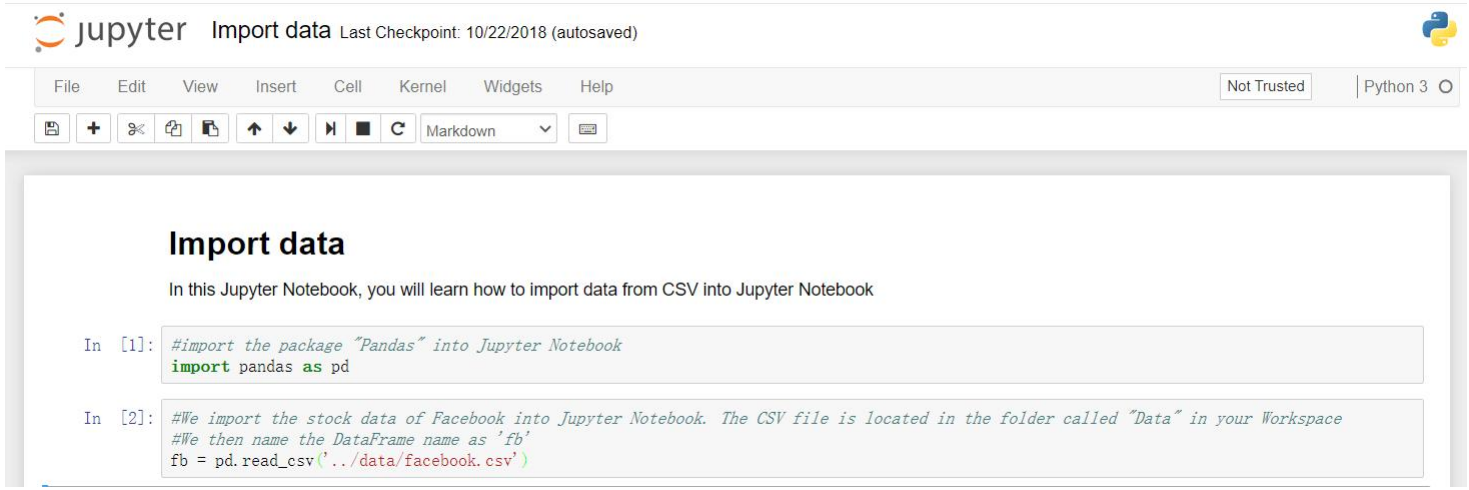
We can use a method type to check the data structure of fb. As we can see, this is a DataFrame from pandas. **By looking at this output**, the historical data of Facebook has been successfully imported as a DataFrame.

Importing data from CSV files into Jupyter Notebook

In this Jupyter Notebook, you will learn how to **import stock data**, which is usually stored in CSV format, into a new DataFrame for doing data analysis.

When using the Jupyter Notebook, be sure you **run the codes sequentially from the first block**, this is to ensure that all packages and corresponding csv files can be installed/imported properly before practicing other codes





Instruction

Now is your turn to import the stock price of Microsoft (microsoft.csv), of which the CSV is located in the same folder, and rename the Dataframe in "ms".

In [3]:
```
ms = pd.read_csv('../data/microsoft.csv')
```

In [4]:
```
# run this cell to ensure Microsoft's stock data is imported
print(ms.iloc[0, 0])
```

```
2014-12-31
```

pd.read_csv or pd.DataFrame.from_csv

pd.read_csv or pd.DataFrame.from_csv

For csv data file loading, pd.DataFram.from_csv is deprecated since version 0.21.0 and now it stops function. **Use pandas.read_csv() instead**. For example

```
fb = pd.read_csv('../data/facebook.csv',index_col=0)
```

"../data/" is the path to the data. You need to tell which column of the csv file you want to set as index, for example, the first column with "index_col=0" . Otherwise, it will use 0,1,2,3..as index by default.

下周上课，将随机抽取同学
进行作业展示

Thank You