第二章 数据化运营的数据来源

- 1、数据来源类型
  - 数据文件
    - txt（任意指定分隔符）、cvs（以逗号分隔的数据文件）、tsv（以tab制表符分隔的数据文件）
  - 数据库
    - 面向高性能并发读写的键值（Key-Value）数据库
      - 优点是具有极高的并发读写性能、查找速度快，典型代表是Redis、Tokyo Cabinet、Voldemort。
    - 面向海量文档的文档数据库
      - 优点是对数据要求不严格，无需提前定义和维护表结构，典型代表为MongoDB、CouchDB
    - 面向可扩展性的列式数据库
      - 优点是查找速度快，可扩展性强，通过分布式扩展来适应数据量的增加以及数据结构的变化，典型代表是Cassandra、HBase、Riak
    - 面向图结构的图形数据库（Graph Database）
      - 优点是利用图结构相关算法，满足特定的数据计算需求，例如最短路径搜寻、关系查询等，典型代表是Neo4J、InfoGrid、Infinite Graph
  - API
    - 服务型API
      - 基于预定义的规则，通过调用API实现特定功能
    - 数据型API
      - 通过特定的语法，通过向服务器发送数据请求，返回特定格式的数据（或数据文件）
  - 流式数据
    - 用户行为数据流
    - 机器数据流
  - 外部公开数据
- 2、使用python获取运营数据
  - 从文本文件
    - 使用read、readline、readlines
      - read
        - 读取文件中的全部数据，直到到达定义的size字节数上限。内容字符串，所有行合并为一个字符串
      - readline
        - 读取文件中的一行数据，直到到达定义的size字节数上限。内容字符串
      - readlines
        - 读取文件中的全部数据，直到到达定义的size字节数上限。内容列表，每行数据作为列表中的一个对象

- 使用Numpy的loadtxt、load、fromfile
  - loadtxt
    - 从txt文本中读取数据。从文件中读取的数组
  - load
    - 从npy、npz或pickled文件加载数组或pickled对象。从数据文件中读取的数组、元组、字典等
  - fromfile
    - 读取简单的文本文件数据以及二进制数据。从文件中读取的数据
- 使用Pandas的read_csv、read_fwf、read_table
  - read_csv
    - 读取csv文件。DataFrame 或TextParser
  - read_fwf
    - 读取表格或固定宽度格式的文本行到数据框。 DataFrame 或TextParser
  - read_table
    - 读取通用分隔符分隔的数据文件到数据框。 DataFrame 或TextParser
- 从excel
  - xls（Excel 97-2003）和xlsx（Excel 2007及以上）
- 从mysql
- 从非关系型数据库MongoDB
- 从API

- 3、读取非结构化数据