Introduction

Chapter 1

Michael Tsiang

Stats 167: Introduction to Databases

UCLA

UCLA

Motivation

Course Overview

Motivation

Where We Came From

Throughout most of the history of statistics, collecting data was costly and time-consuming.

Even today, in many application areas (e.g., psychology, biology, or medicine), data collection still requires considerable time and resources, and it often results in a dataset with a handful of observations.

Classical statistical methods have emphasized the problems that arise in **sample-starved** (or small sample) scenarios.

For example, the Student's t-distribution and Fisher's exact test are statistical tools developed for when asymptotic results (usually the Central Limit Theorem) are not applicable.

The Problem With Big Data

In recent decades, collecting data in many scenarios has had the opposite issue, leading to the term **big data**, which refers to data that is too large and complex to be analyzed with classical statistical methods.

For example, social media sites and streaming platforms collect data on a myriad of metrics on every active user in real time, generating gigabytes of data constantly. The **volume**, **velocity**, and **variety** of big data have created problems that classical statisticians could not have dreamed of, such as:

- How does one store gigabytes (or even terabytes) of data that come in every second?
- How do we access only the information we need from the flood of data?
- How do we make sense of the data to make good predictions for the future (e.g., consumer behavior, movie recommendations, profit projections, etc.)?

The Classroom Setting

In most statistics classes, sample datasets are typically provided in relatively small rectangular tables, where each row is an observation and each column is a variable (i.e., *tidy* data).

These datasets are stored in standard file formats (like `.csv`), and they are small enough to be easily downloaded and stored on a local computer. The data is also usually pre-cleaned, so there are no errors in data entry, and missing values are both sparse and properly coded.

Usually, a simple `readr::read_csv()` or `pandas.read_csv()` call will import the data into R or Python and be easily viewable at a glance.

Beyond The Classroom

Since we never encounter big data in the classroom, then we never need to consider the problems that arise from it.

Issues of storage capacity, data management, computation time, or storage and computational efficiency are theoretical considerations in a classroom setting, but they are not noticeable for nearly every example when you are implementing methods on such limited datasets.

What happens when we no longer have 200 observations? What if we have 200 million observations? What happens when we no longer have 10 variables? What if we have 10,000 variables?

Beyond The Rectangle

While a single table of data is useful for focusing on learning statistical and machine learning methods, it is an unrealistic view of how data is stored and accessed in many practical applications.

- What happens when collected data does not fit well into a tabular form (e.g., text, images, PDFs)?
- What happens when data becomes too big for a single laptop's memory (e.g., millions of observations and/or thousands of variables)?
- What happens if multiple teams (e.g., sales, purchasing, analytics) need to access and update the data?

The single table structure does not scale well to larger data (both in size and scope). In this more practical and modern context, we need to move beyond the concept of a small single rectangular dataset to a database.

Course Overview

Populer... lar

A **database** is an organized collection of data.

The most common type of database is the **relational database**, which stores data in a collection of related tables.

The universal language for relational databases is **Structured Query Language (SQL)**, which will be a primary (but not only) focus of this course.

SQL is one of the most popular requirements for data science jobs.

Two KD Nuggets articles:

- Why SQL is THE Language to Learn for Data Science
- Why SQL Will Remain the Data Scientist's Best Friend

While SQL is not the only tool you need in your data scientist toolkit, it is a fundamentally important one for many careers in data engineering, data analytics, and data science.

Where We're Going

To put relational databases and SQL into context, we will first introduce a few common concepts of data management and data engineering.

The course outline of topics:

- Basics of Data Management
- Data Engineering concepts
- SQL (Basic to Intermediate)
- NoSQL Databases (e.g., Key-value, Document, Graph)

The goal of the course is to provide intuition on fundamental concepts of databases and hands-on opportunity to practice building skills in working with databases to serve as practical preparation for applications, interviews, and careers in data science.