Data Transformation

Data Transformation

- **Why you might wish to convert file formats prior to analysis**
- **How you can join both small and large data sets**
- **What anonymization is and why it's important**
- **How re-identification can expose an organization to liability**

Data Transformation

- **File format conversion**
- Joining data sets
- Anonymization

File Format Conversion

- **Sometimes data isn't provided in the same format you require**
  - The format might be suitable for data collection but not analysis
  - It might not be appropriate at expected scale
  - It might not be supported by the tool you need
  - Another format might offer better performance
  - Another format might be better for long-term storage
- **The solution is often to convert data to another format**

Brief Introduction to Apache Hive

- **Another way of converting file formats involves using Apache Hive**
  - Let's first briefly cover what Hive is and what it can do
- **Hive is an alternative to writing low-level MapReduce code**
  - Users can analyze data stored in Hadoop data via HiveQL
    - HiveQL is a declarative language very similar to SQL
- **Hive does *not* turn your Hadoop cluster into a database**
  - Instead, the Hive interpreter turns HiveQL into MapReduce jobs
  - Hive tables are simply directories of data stored in HDFS
    - The `create table` statement instructs Hive how to parse it

Joining Data Sets with Hive

- **Hive is an alternative to writing low-level MapReduce code**
- **Joining data sets with Hive is easy**
  - Usually preferable to writing MapReduce code to do joins
- **Benefits of using Hive for joins**
  - Far less code
  - Much quicker to write
  - Less chance for error
  - Requires far less skill, so it's accessible to more people
- **Disadvantages of using Hive**
  - Slightly less control