**Motivation**

- Computational and data-driven approaches have long been a standard method for scientific experimentation.
- Growing number of fields/industries depend on computational experiments, e.g., social sciences, data analytics.
- Despite the increasing importance, systematically organizing the data for/from experimentation and maintaining the infrastructure for running such experiments are still tedious and difficult if not painful.

**Typical Difficulties**

- Maintaining improvised scripts and output formats
- Producing ad-hoc summaries and charts from results
- Executing runs/re-runs selectively, in parallel/distributed
- Keeping track of which code produced particular output
- Organizing parameter-dependent data

**Goal**

- Enable experimenters to make discoveries more quickly by reducing the burden of data and execution management.

**Overview**

- **Experiment Code**
  - e.g., Sorting algorithm implementations
- **Input Parameters to try**
  - Algo: Bubble sort, Quicksort, ...
  - DataSize: small, medium, large
  - DataOrder: random, sorted, reversed
- **Desired Outputs**
  - Time taken to sort
  - Number of array accesses

**Features**

- Standardized Repository
  - Factorization of code and data
  - Efficient use of storage by data de-duplication
- Execution Planning and Management
  - Compute resource abstraction: local, remote, cluster
- Immediate, Interactive Visualization
  - Data provenance: drill-down individual runs of aggregate data

**Demo**

- Comparing Sorting Algorithms
- Giant Components in Random Networks

**User Case Study**

- Bongsoo Suh (Baccus Lab, Neurobiology Department)
  Computational modeling of retinal contrast adaptation.
  Model optimizations with MATLAB on FarmShare cluster.

**Work in Progress**

**AutoRunner**

Given a budget (e.g., number of runs, time), automatically execute runs to augment the current result dataset for:

- Confirming correlation between inputs and outputs
- Discovering interaction effects between inputs
- Reducing the variation in output of repeated runs

**VizRec (Visualization Recommender)**

Given a result dataset, automatically find a set of visualizations that show interesting aspects of the experiment result.

**Evolution of Experiments**

Maintain multiple versions of experiments as constant change to code and input/output data are integral part of exploratory experiments. Identify and share the subset of result dataset that stays compatible between different versions.