Performance Evaluation:

Introduction to Experiment Design

Hongwei Zhang

http://www.cs.wayne.edu/~hzhang

The first ninety percent of the task takes ten percent of the time, and the last ten percent takes the other ninety percent.

--- Ninety-ninety rule of project schedules

Acknowledgement: this lecture is partially based on the slides of Dr. Raj Jain.

Experiment design and analysis

- Performance analysis often requires that the *effects of individual factors be identified*, so that meaningful statements can be made about *different levels of each factor* (e.g., different systems or network protocols)

- Objective of experiment design (in measurement and simulation)?
  - To obtain the maximum information with the minimum number of experiments

Experiment design and analysis (contd.)

- Experiment analysis
  - Develop a model that best describes the data obtained
    - Regression models (Chapters 14&15), modeling categorical data (Chapters 17-23)
    - Estimate confidence intervals for model parameters
  - Estimate the contribution of each factor to performance;
  - Check if the alternatives of a factor are significantly different in their impact.

Example

- Personal workstation design
  - Processor: 68000, Z80, or 8086 (the old days ☺)
  - Memory size: 512K, 2M, or 8M bytes
  - Number of Disks: One, two, three, or four
  - Workload: Secretarial, managerial, or scientific
  - User education: High school, college, or postgraduate level

  - Five **Factors** at 3x3x4x3x3 **levels**

Outline

- Terminology
- Common mistakes
- Sample experimental design methods

Outline

- Terminology
- Common mistakes
- Sample experimental design methods

Terminology

- **Response Variable**: outcome of an experiment
  - E.g., throughput, response time
- **Factors**: Variables that affect the response variable
  - E.g., CPU type, memory size, number of disk drives, workload used, and user's educational level

  Also called *predictor variables* or *predictors*
- **Levels**: The values that a factor can assume
  - E.g., the CPU type has three levels: 68000, 8080, or Z80; # of disk drives has four levels.

  Also called *treatment*
- **Primary Factors**: The factors whose effects need to be quantified
  - E.g., CPU type, memory size only, and number of disk drives

Terminology (contd.)

- **Secondary Factors**: Factors whose impact need not be quantified
  - E.g., the workloads and user educational level
- **Replication**: Repetition of all or some experiments
- **Design**: a design specifies the # of experiments, the factor level and the # of replications for each experiment
  - E.g., Full Factorial Design with 5 replications: 3*3*4*3*3 or 324 experiments, each repeated five times.
- **Experimental Unit**: Any entity that is used for experiments
  - E.g., users
  - Generally, no interest in comparing the units;

    Goal - minimize the impact of variation among the units.

Terminology (contd.)

- **Interaction:** effect of one factor depends upon the level of the other

Table 1: Noninteracting Factors

	A_1	A_2
B_1	3	5
B_2	6	8

Impact of factor A is always 2 irrespective of the level of factor B

Table 2: Interacting Factors

	A_1	A_2
B_1	3	5
B_2	6	9

Impact of factor A depends on the level of factor B

Outline

- Terminology
- Common mistakes
- Sample experimental design methods

Common mistakes in experimentation

- Variation due to experimental errors is ignored
  - Variation due to a factor must be compared with that due to errors before making a decision about its effect
- Important parameters are not controlled
  - E.g., the right user population depends on application scenarios
- Effects of different factors are not isolated
  - E.g., change all factors in one experiment
- Simple one-factor-at-a-time designs are used
  - May well be wasteful of resources
  - Interactions may not be identified
- Too many experiments are conducted.
  - Better: two phases (many factors + few levels => few factors + many levels)

Outline

- Terminology
- Common mistakes
- Sample experimental design methods

Three most commonly used experiment designs

- **Simple Designs**: start with a typical configuration, and vary one factor at a time

$$\#\text{ of Experiments} = 1 + \sum_{i=1}^{k} (n_i - 1)$$

  - (-) Not statistically efficient
  - (-) Lead to wrong conclusions if factors have interaction
  - *Not recommended*

- **Full Factorial Design**: All combinations.

$$\#\text{ of Experiments} = \prod_{i=1}^{k} n_i$$

  - (+) Can find the effect of all factors
  - (-) Too much time and money
  - *May try 2^k design first*

Commonly used experiment design (contd.)

- Fractional Factorial Designs: only use a fraction of the full factorial design
  - (+) Save time and expense
  - (-) Less information
    - e.g., may not be able to identify interactions among all factors; Not a problem if those interactions are negligible.

An example of fractional factorial design

- Workstation design
  - Ignoring the # of disk drives, a full factorial design needs

$$(3\text{ CPUs})(3\text{ Memory levels})(3\text{ workloads})(3\text{ ed levels}) \\ = 81\text{ experiments}$$

  - A fractional factorial design may only need 9 experiments

Experiment Number	CPU	Memory Level	Workload Type	Educational Level
1	68000	512K	Managerial	High School
2	68000	2M	Scientific	Post-graduate
3	68000	8M	Secretarial	College
4	Z80	512K	Scientific	College
5	Z80	2M	Secretarial	High School
6	Z80	8M	Managerial	Post-graduate
7	8086	512K	Secretarial	Post-graduate
8	8086	2M	Managerial	College
9	8086	8M	Scientific	High School

Note that each of the four factors is used three times at each of its three levels

Summary

- Terminology
- Common mistakes
- Sample experimental design methods

Exercise

The performance of a system depends upon the following three factors:

- CPU type: 68000, 8086, 80286
- Operating System type: CPM, MS-DOS, UNIX
- Disk drive type: A, B, C

How many experiments are required to analyze the performance if

- There is significant interaction among factors;
- There is no interaction among factors; or
- The interactions are small compared to main effects.