CS533
Modeling and Performance Evaluation of Network and Computer Systems

Experimental Design

(Chapters 16-17)

Introduction (1 of 3)

No experiment is ever a complete failure. It can always serve as a negative example. – Arthur Bloch

The fundamental principle of science, the definition almost, is this: the sole test of the validity of any idea is experiment. – Richard P. Feynman

- Goal is to obtain maximum information with minimum number of experiments
- Proper analysis will help separate out the factors
- Statistical techniques will help determine if differences are caused by variations from errors or not

2

Introduction (2 of 3)

- Key assumption is non-zero cost
  - Takes time and effort to gather data
  - Takes time and effort to analyze and draw conclusions

  → Minimize number of experiments run
- Good experimental design allows you to:
  - Isolate effects of each input variable
  - Determine effects due to interactions of input variables
  - Determine magnitude of experimental error
  - Obtain maximum info with minimum effort

3

Introduction (3 of 3)

- Consider
  - Vary one input while holding others constant
    - Simple, but ignores possible interaction between two input variables
  - Test all possible combinations of input variables
    - Can determine interaction effects, but can be very large
    - Ex: 5 factors with 4 levels → 4^5 = 1024 experiments. Repeating to get variation in measurement error 1024x3 = 3072
- There are, of course, in-between choices…
  - (Ch 19, but leads to confounding…)

4

Outline

- Introduction
- Terminology
- General Mistakes
- Simple Designs
- Full Factorial Designs
  - 2^k Factorial Designs
- $2^k r$ Factorial Designs

5

Terminology (1 of 4)

(Will explain terminology using example)
- Study PC performance
  - CPU choice: 6800, z80, 8086
  - Memory size: 512 KB, 2 MB, 8 MB
  - Disk drives: 1-4
  - Workload: secretarial, managerial, scientific
  - Users: high school, college, graduate
- *Response variable* – the outcome or the measured performance
  - Ex: throughput in tasks/min or response time for a task in seconds

6

Terminology (2 of 4)

- *Factors* – each variable that affects response
  - Ex: CPU, memory, disks, workload, user
  - Also called *predictor variables* or *predictors*
- *Levels* – the different values factors can take
  - EX: CPU 3, memory 3, disks 4, workload 3, users 3
  - Also called *treatment*
- *Primary factors* – those of most important interest
  - Ex: maybe CPU and memory the most

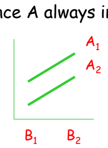
Terminology (3 of 4)

- *Secondary factors* – of less importance
  - Ex: maybe user type not as important
- *Replication* – repetition of all or some experiments
  - Ex: if run three times, then three replications
- *Design* – specification of the replication, factors, levels
  - Ex: Specify all factors, at above levels with 5 replications so 3x3x4x3x3 = 324 time 5 replications yields 1215 total

Terminology (4 of 4)

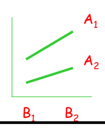
- *Interaction* – two factors A and B interact if one shows dependence upon another
  - Ex: non-interacting factor since A always increases by 2

	A_1	A_2
B_1	3	5
B_2	6	8



  - Ex: interacting factors since A change depends upon B

	A_1	A_2
B_1	3	5
B_2	6	9



Outline

- Introduction
- Terminology
- General Mistakes
- Simple Designs
- Full Factorial Designs
  - 2^k Factorial Designs
- 2^kr Factorial Designs

Common Mistakes in Experiments (1 of 2)

- *Variation due to experimental error is ignored.*
  - Measured values have randomness due to measurement error. Do not assign (or assume) all variation is due to factors.
- *Important parameters not controlled.*
  - All parameters (factors) should be listed and accounted for, even if not all are varied.
- *Effects of different factors not isolated.*
  - May vary several factors simultaneously and then not be able to attribute change to any one.
  - Use of simple designs (next topic) may help but have their own problems.

Common Mistakes in Experiments (2 of 2)

- *Interactions are ignored.*
  - Often effect of one factor depend upon another. Ex: effects of cache may depend upon size of program. Need to move beyond one-factor-at-a-time designs
- *Too many experiments are conducted.*
  - Rather than running all factors, all levels, at all combinations, break into steps
  - First step, few factors and few levels
    - Determine which factors are significant
    - Two levels per factor (details later)
  - More levels added at later design, as appropriate

Outline

- Introduction
- Terminology
- General Mistakes
- Simple Designs
- Full Factorial Designs
  - 2^k Factorial Designs
- $2^k r$ Factorial Designs

13

Simple Designs

- Start with typical configuration
- Vary one factor at a time
- Ex: typical may be PC with z80, 2 MB RAM, 2 disks, managerial workload by college student
  - Vary CPU, keeping everything else constant, and compare
  - Vary disk drives, keeping everything else constant, and compare
- Given k factors, with ith having n_i levels

  Total = $1 + \Sigma(n_i-1)$ for $i = 1$ to k
- Example: in workstation study

  1 + (3-1) + (3-1) + (4-1) + (3-1) + (3-1) + (3-1) = 14
- But may ignore interaction

  (Example next)

14

Example of Interaction of Factors

- Consider response time vs. memory size and degree of multiprogramming

Degree	32 MB	64 MB	128MB
1	0.25	0.21	0.15
2	0.52	0.45	0.36
3	0.81	0.66	0.50
4	1.50	1.45	0.70

- If fixed degree 3, mem 64 and vary one at a time, may miss interaction
  - Example: degree 4, non-linear response time with memory

15

Outline

- Introduction
- Terminology
- General Mistakes
- Simple Designs
- Full Factorial Designs
  - 2^k Factorial Designs
- $2^k r$ Factorial Designs

16

Full Factorial Designs

- Every possible combination at all levels of all factors
- Given k factors, with ith having n_i levels

  Total = $\Pi\ n_i$ for $i = 1$ to k
- Example: in CPU design study

  (3 CPUs)(3 mem) (4 disks) (3 loads) (3 users)

  = 324 experiments
- Advantage is can find every interaction component
- Disadvantage is costs (time and money), especially since may need multiple iterations (later)
- Can reduce costs by: reduce levels, reduce factors, run fraction of full factorial

  (Next, reduce levels)

17

2^k Factorial Designs

Twenty percent of the jobs account for 80% of the resource consumption.
– Pareto's Law

- Very often, many levels at each factor
  - Ex: effect of network latency on user response time → there are lots of latency values to test
- Often, performance continuously increases or decreases over levels
  - Ex: response time always gets higher
  - Can determine direction with min and max
- For each factor, choose 2 alternatives at each level
  - 2^k factorial designs
- Then, can determine which of the factors impacts performance the most and study those further

18

2^2 Factorial Design (1 of 4)

- Special case with only 2 factors
  - Easily analyzed with regression
- Example: MIPS for Mem (4 or 16 Mbytes) and Cache (1 or 2 Kbytes)

	Mem 4MB	Mem 16MB
Cache 1 KB	15	45
Cache 2 KB	25	75

- Define x_a = -1 if 4 Mbytes mem, +1 if 16 Mbytes
- Define x_b = -1 if 1 Kbyte cache, +1 if 2 Kbytes
- Performance:

$$y = q_0 + q_a x_a + q_b x_b + q_{ab} x_a x_b$$

2^2 Factorial Design (2 of 4)

- Substituting:

$$15 = q_0 - q_a - q_b + q_{ab}$$
$$45 = q_0 + q_a - q_b - q_{ab}$$
$$25 = q_0 - q_a + q_b - q_{ab}$$
$$75 = q_0 + q_a + q_b + q_{ab}$$

(4 equations in 4 unknowns)

- Can solve to get:

$$y = 40 + 20x_a + 10x_b + 5x_a x_b$$

- Interpret:
  - Mean performance is 40 MIPS, memory effect is 20 MIPS, cache effect is 10 MIPS and interaction effect is 5 MIPS

(Generalize to easier method next)

2^2 Factorial Design (3 of 4)

Exp	a	b	y
1	-1	-1	y_1
2	1	-1	y_2
3	-1	1	y_3
4	1	1	y_4

$$y = q_0 + q_a x_a + q_b x_b + q_{ab} x_a x_b$$

- So:

$$y_1 = q_0 - q_a - q_b + q_{ab}$$
$$y_2 = q_0 + q_a - q_b - q_{ab}$$
$$y_3 = q_0 - q_a + q_b - q_{ab}$$
$$y_4 = q_0 + q_a + q_b + q_{ab}$$

- Solving, we get:

$$q_0 = \tfrac{1}{4}(y_1 + y_2 + y_3 + y_4)$$
$$q_a = \tfrac{1}{4}(-y_1 + y_2 - y_3 + y_4)$$
$$q_b = \tfrac{1}{4}(-y_1 - y_2 + y_3 + y_4)$$
$$q_{ab} = \tfrac{1}{4}(y_1 - y_2 - y_3 + y_4)$$

- Notice for q_a can obtain by multiplying "a" column by "y" column and adding
  - Same is true for q_b and q_{ab}

2^2 Factorial Design (4 of 4)

i	a	b	ab	y
1	-1	-1	1	15
1	1	-1	-1	45
1	-1	1	-1	25
1	1	1	1	75
160	80	40	20	Total
40	20	10	5	Ttl/4

- Column "i" has all 1s
- Columns "a" and "b" have all combinations of 1, -1
- Column "ab" is product of column "a" and "b"
- Multiply column entries by y_i and sum
- Dived each by 4 to give weight in regression model
- Final:

$$y = 40 + 20x_a + 10x_b + 5x_a x_b$$

Allocation of Variation (1 of 3)

- Importance of a factor measured by proportion of total variation in response explained by the factor
  - Thus, if two factors explain 90% and 5% of the response, then the second may be ignored
    - Ex: capacity factor (768 Kbps or 10 Mbps) versus TCP version factor (Reno or Sack)
- Sample variance of y

$$s_y^2 = \Sigma(y_i - \bar{y})^2 / (2^2 - 1)$$

- With numerator being total variation, or Sum of Squares Total (SST)

$$SST = \Sigma(y_i - \bar{y})^2$$

Allocation of Variation (2 of 3)

- For a 2^2 design, variation is in 3 parts:
  - $SST = 2^2 q^2_a + 2^2 q^2_b + 2^2 q^2_{ab}$ (Derivation 17.1, p.287)
- Portion of total variation:
  - of a is $2^2 q^2_a$
  - of b is $2^2 q^2_b$
  - of ab is $2^2 q^2_{ab}$
- Thus, SST = SSA + SSB + SSAB
- And fraction of *variation* explained by a:

= SSA/SST

  - Note, may not explain the same fraction of *variance* since that depends upon errors+

Allocation of Variation (3 of 3)

- In the memory-cache study
  $\bar{y} = \frac{1}{4}(15 + 55 + 25 + 75) = 40$
- Total variation
  $= \Sigma(y_i - \bar{y})^2 = (25^2 + 15^2 + 15^2 + 35^2)$
  $= 2100 = 4\text{x}20^2 + 4\text{x}10^2 + 4\text{x}5^2$
- Thus, total variation is 2100
  - 1600 (of 2100, 76%) is attributed to memory
  - 400 (of 2100, 19%) is attributed to cache
  - Only 100 (of 2100, 5%) is attributed to interaction
- This data suggests exploring memory further and not spending more time on cache (or interaction)

(That was for 2 factors. Extend to k next)

25

General 2^k Factorial Designs (1 of 4)

- Can extend same methodology to k factors, each with 2 levels → Need 2^k experiments
  - k main effects
  - (k choose 2) two factor effects
  - (k choose 3) three factor effects…
- Can use sign table method

(Show with example, next)

26

General 2^k Factorial Designs (2 of 4)

- Example: design LISP machine
  - Cache, memory and processors

Factor	Level -1	Level 1
Memory (a)	4 Mbytes	16 Mbytes
Cache (b)	1 Kbytes	2 Kbytes
Processors (c)	1	2

- The 2^3 design and MIPS perf results are:

	4 Mbytes Mem(a)		16 Mbytes Mem	
Cache (b)	One proc (c)	Two procs	One proc	Two procs
1 KB	14	46	22	58
2 KB	10	50	34	86

27

General 2^k Factorial Designs (3 of 4)

- Prepare sign table:

i	a	b	c	ab	ac	bc	abc	y
1	-1	-1	-1	1	1	1	-1	14
1	1	-1	-1	-1	-1	1	1	22
1	-1	1	-1	1	-1	-1	-1	10
1	1	1	-1	1	-1	-1	-1	34
1	-1	1	1	-1	-1	1	-1	46
1	1	-1	1	-1	1	-1	-1	58
1	-1	1	1	-1	-1	1	-1	50
1	1	1	1	1	1	1	1	86
320	80	40	160	40	16	24	9	Ttl
40	10	5	20	5	2	3	1	Ttl/8

$q_a = 10$, $q_b = 5$, $q_c = 20$ and $q_{ab} = 5$, $q_{ac} = 2$, $q_{bc} = 3$ and $q_{abc} = 1$

28

General 2^k Factorial Designs (3 of 4)

- $q_a=10$, $q_b=5$, $q_c=20$ and $q_{ab}=5$, $q_{ac}=2$, $q_{bc}=3$ and $q_{abc}=1$
- $SST = 2^3 (q_a^2+q_b^2+q_c^2+q_{ab}^2+q_{ac}^2+q_{bc}^2+q_{abc}^2)$
  $= 8\,(10^2+5^2+20^2+5^2+2^2+3^2+1^2)$
  $= 800+200+3200+200+32+72+8$
  $= 4512$
- The portion explained by the 7 factors are:
  - mem = 800/4512 (18%)
  - cache = 200/4512 (4%)
  - proc = 3200/4512 (71%)
  - mem-cache =200/4512 (4%)
  - mem-proc = 32/4512 (1%)
  - cache-proc = 72/4512 (2%)
  - mem-proc-cache = 8/4512 (0%)

29

Outline

- Introduction
- Terminology
- General Mistakes
- Simple Designs
- Full Factorial Designs
  - 2^k Factorial Designs
- 2^kr Factorial Designs

30

$2^k r$ Factorial Designs

> *No amount of experimentation can ever prove me right; a single experiment can prove me wrong.* -Albert Einstein

- With 2^k factorial designs, not possible to estimate error since only done once
- So, repeat r times for $2^k r$ observations
- As before, will start with $2^2 r$ model and expand
- Two factors at two levels and want to isolate experimental errors
  - Repeat 4 configurations r times
- Gives you error term:
  - $y = q_0 + q_a x_a + q_b x_b + q_{ab} x_a x_b + e$
  - Want to quantify e

(Illustrate by example, next)

$2^2 r$ Factorial Design Errors (1 of 2)

- Previous cache experiment with r=3

i	a	b	ab	y	mean y
1	-1	-1	1	(15, 18, 12)	15
1	1	-1	-1	(45, 48, 51)	48
1	-1	1	-1	(25, 28, 19)	24
1	1	1	1	(75, 75, 81)	77
164	86	38	20		Total
41	21.5	9.5	5		Ttl/4

- Have estimate for each y
  - $\hat{y}_i = q_0 + q_a x_{ai} + q_b x_{bi} + q_{ab} x_{ai} x_{bi} + e_i$
- Have difference (error) for each repetition
  - $e_{ij} = y_{ij} - \hat{y}_i = y_{ij} - q_0 - q_a x_{ai} - q_b x_{bi} - q_{ab} x_{ai} x_{bi}$

$2^2 r$ Factorial Design Errors (2 of 2)

- Use sum of squared errors (SSE) to compute variance and confidence intervals

$$SSE = \Sigma\Sigma e^2_{ij} \text{ for } i = 1 \text{ to } 4 \text{ and } j = 1 \text{ to } r$$

- Example

i	a	b	ab	$\hat{y}_i$	y_{i1} y_{i2} y_{i3}	e_{i1} e_{i2} e_{i3}
1	-1	-1	1	15	15 18 12	0 3 -3
1	1	-1	-1	48	45 48 51	-3 0 3
1	-1	1	-1	24	25 28 19	1 4 -5
1	1	1	1	77	75 75 81	-2 -2 4

- Ex: $\hat{y}_1 = q_0 - q_a - q_b + q_{ab} = 41 - 21.5 - 9.5 + 5 = 15$
- Ex: $e_{11} = y_{11} - \hat{y}_1 = 15 - 15 = 0$
- $SSE = 0^2+3^2+(-3)^2+(-3)^2+0^2+3^2+1^2+4^2+(-5)^2 +(-2)^2+(-2)^2+4^2 = 102$

$2^2 r$ Factorial Allocation of Variation

- Total variation (SST)

$$SST = \Sigma(y_{ij} - \bar{y}_{..})^2$$

- Can be divided into 4 parts:

$$\Sigma(y_{ij} - \bar{y}_{..})^2 = 2^2 r q^2_a + 2^2 r q^2_b + 2^2 r q^2_{ab} + \Sigma e^2_{ij}$$

$$SST = SSA + SSB + SSAB + SSE$$

- Thus
  - SSA, SSB, SSAB are variations explained by factors a, b and ab
  - SSE is unexplained variation due to experimental errors
- Can also write SST = SSY-SS0 where SS0 is sum squares of mean

(Derivation 18.1, p.296)

$2^2 r$ Factorial Allocation of Variation Example

- For memory cache study:
  - $SSY = 15^2+18^2+12^2+ \ldots +75^2 + 81^2 = 27{,}204$
  - $SS0 = 2^2 r q^2_0 = 12 \times 41^2 = 20{,}172$
  - $SSA = 2^2 r q^2_a = 12 \times (21.5)^2 = 5547$
  - $SSB = 2^2 r q^2_b = 12 \times (9.5)^2 = 1083$
  - $SSAB = 2^2 r q^2_{ab} = 12 \times 5^2 = 300$
  - $SSE = 27{,}204 - 2^2 \times 3(41^2+21.5^2+9.5^2+5^2) = 102$
  - $SST = 5547 + 1083 + 300 + 102 = 7032$
- Thus, total variation of 7032 divided into 4 parts:
  - Factor a explains 5547/7032 (78.88%), b explains 15.40%, ab explains 4.27%
  - Remaining 1.45% unexplained and attributed to error

Confidence Intervals for Effects

- Assuming errors are normally distributed, then y_{ij}s are normally distributed with same variance
- Since q_o, q_a, q_b, q_{ab} are all linear combinations of y_{ij}'s (divided by $2^2 r$), then they have same variance (divided by $2^2 r$)
- Variance $s^2 = SSE / (2^2(r-1))$
- Confidence intervals for effects then:
  - $q_i \pm t_{[1-\alpha/2;\, 2^2(r-1)]} s_{qi}$
- If confidence interval does not include zero, then effect is significant

Confidence Intervals for Effects (Example)

- Memory-cache study, std dev of errors:
  s_e = sqrt[SSE / (2^2(r-1)] = sqrt(102/8) = 3.57
- And std dev of effects:
  $s_{qi} = s_e$ / sqrt(2^2r) = 3.57/3.47 = 1.03
- The t-value at 8 degrees of freedom and 95% confidence is 1.86
- Confidence intervals for parameters:
  $q_i \pm(1.86)(1.03) = q_i \pm 1.92$
  - q_0 → (39.08,42.91), q_a→(19.58,23,41), q_b→(7.58,11.41), q_{ab}→(3.08,6.91)
  - Since none include zero, all are statistically significant

37

Confidence Intervals for Predicted Responses (1 of 2)

- Mean response predicted
  - $y = q_0 + q_a x_a + q_b x_b + q_{ab} x_a x_b$
- If predict mean from *m* more experiments, will have same mean but confidence interval on predicted response decreases
- Can show that std dev of predicted y with me more experiments
  - $s_{ym} = s_e$sqrt($1/n_{eff}$ + 1/m)
  - Where n_{eff} = runs/(1+df)
    - In 2 level case, each parameter has 1 df, so $n_{eff} = 2^2r/5$

38

Confidence Intervals for Predicted Responses (2 of 2)

- A 100(1-α)% confidence interval of response:
  - $y_p \pm t_{[1-\alpha/2;\ 2^2(r-1)]} s_{ym}$
- Two cases are of interest.
  - Std dev of one run (m=1)
    - $s_{y1} = s_e$sqrt($5/2^2r$ + 1)
  - Std dev for many runs (m=∞)
    - $s_{y1} = s_e$sqrt($5/2^2r$)

39

Confidence Intervals for Predicted Responses Example (1 of 2)

- Mem-cache study, for x_a=-1, x_b=-1
- Predicted mean response for future experiment
  - $y_1 = q_0 - q_a - q_b + q_{ab}$ = 41-21.5+1=15
  - Std dev = 3.57 x sqrt(5/12 + 1) = 4.25
- Using t[0.95;8] = 1.86, 90% conf interval
  15±1.86x4.25 = (8.09,22.91)
- Predicted mean response for 5 future experiments
  - Std dev = 3.57(sqrt 5/12 + 1/5) = 2.80
    15±1.86x2.80 = (9.79,20.29)

40

Confidence Intervals for Predicted Responses Example (2 of 2)

- Predicted Mean Response for Large Number of Experiments
  - Std dev = 3.57xsqrt(5/12) = 2.30
  - The confidence interval:
    15±1.86x2.30=(10.72,19.28)

41