DF: design factor
NF: nuisance factor

Latin Square Designs

- Until now we have discussed experimental designs that employ blocking to control for *one* nuisance factor
  - If we want to control for *two* nuisance factors, we should use a **Latin square design**
  - If we want to control for *three* nuisance factors, we should use a **Graeco-Latin square design**
  - If we want to control for *four* nuisance factors, we should use a **Hyper-Graeco-Latin square design**

- A Latin square of order p is a $p \times p$ grid containing p unique symbols
  - Each of these symbols occurs exactly once in each column
  - Each of these symbols occurs exactly once in each row
  - These "symbols" are typically denoted by Latin letters

A	C	B
C	B	A
B	A	C

A	B	C	D
C	D	A	B
B	C	D	A
D	A	B	C

A	B	C	D	E
E	A	B	C	D
D	E	A	B	C
C	D	E	A	B
B	C	D	E	A

- A Sudoku puzzle is a special example of a 9×9 Latin square

4	7	6	1	8	3	5	9	2
5	1	3	2	9	7	8	6	4
2	9	8	5	4	6	1	7	3
7	3	4	9	6	1	2	8	5
6	8	5	3	7	2	9	4	1
9	2	1	4	5	8	6	3	7
1	6	9	7	2	4	3	5	8
8	4	2	6	3	5	7	1	9
3	5	7	8	1	9	4	2	6

- We exploit this combinatorial structure to help us design experiments that facilitate blocking by two nuisance factors
  - We arbitrarily associate the p rows with the levels of the first nuisance factor
  - We arbitrarily associate the p columns with the levels of the second nuisance factor
  - We arbitrarily associate the p Latin letters with the levels of the design factor
- Here is an example with $p = 4$

		Nuisance Factor 2			
		1	2	3	4
Nuisance Factor 1	1	A	B	C	D
	2	D	A	B	C
	3	C	D	A	B
	4	B	C	D	A

(3,2) element:
DF at level D
NF1 at level 3
NF2 at level 2

- Each cell in this table represents a "block" in which the nuisance factors' levels are held fixed, and the Latin letter indicates which experimental condition is being executed
- Rows, columns and letters are all orthogonal, allowing us to separately estimate the effects of the design factor and each of the two nuisance factors.
- These effects may be informally summarized with the overall average and level-specific averages of the response variables

$$\overline{y}_{\cdot j \cdot \cdot} = \frac{1}{np} \sum_{(j,k,l) \in \mathcal{S}_j} \sum_{i=1}^{n} y_{ijkl}$$

$$\overline{y}_{\cdot \cdot k \cdot} = \frac{1}{np} \sum_{(j,k,l) \in \mathcal{S}_k} \sum_{i=1}^{n} y_{ijkl}$$

$$\overline{y}_{\cdot \cdot \cdot l} = \frac{1}{np} \sum_{(j,k,l) \in \mathcal{S}_l} \sum_{i=1}^{n} y_{ijkl}$$

$$\overline{y}_{\cdot \cdot \cdot \cdot} = \frac{1}{N} \sum_{(j,k,l) \in \mathcal{S}} \sum_{i=1}^{n} y_{ijkl}$$

- A comment about notation:
  - Each block contains just one condition, so each pair (k, l) uniquely determines the value of j
  - Consequently, there exist just p^2 tuples (j, k, l)
  - Denote them by the set $\mathcal{S}$

(1,1,1)	(2,1,2)	(3,1,3)	(4,1,4)
(4,2,1)	(1,2,2)	(2,2,3)	(3,2,4)
(3,3,1)	(4,3,2)	(1,3,3)	(2,3,4)
(2,4,1)	(3,4,2)	(4,4,3)	(1,4,4)

  - We also define:
    - $\mathcal{S}_j \subset \mathcal{S}$: all tuples for which the design factor level is j
    - $\mathcal{S}_k \subset \mathcal{S}$: all tuples for which nuisance factor 1's level is k
    - $\mathcal{S}_l \subset \mathcal{S}$: all tuples for which nuisance factor 2's level is l

$\mathcal{S}_{j=1} = \{(1,1,1), (1,2,2), (1,3,3), (1,4,4)\}$

- The primary analysis goal in a LSD is to determine whether the expected response differs significantly from one condition to another
  - and if so, to identify the optimal condition
  - while controlling for the potential effect of the nuisance factors

- We've previously done this with gatekeeper tests of the form

$$H_0\text{: } \theta_1 = \theta_2 = \cdots = \theta_p \text{ vs. } H_A\text{: } \theta_j \neq \theta_{j'} \text{ for some } j' \neq j$$

- We do the same thing here, while accounting for the nuisance factors, with *appropriately defined* linear or logistic regression models which contain
  - an intercept,
  - $p-1$ indicator variables for the design factor's levels, and
  - $p-1$ indicator variables for nuisance factor 1's levels, and
  - $p-1$ indicator variables for nuisance factor 2's levels

- We write the linear predictor as

$$\alpha + \sum_{j=1}^{p-1} \beta_j x_{ij} + \sum_{k=1}^{p-1} \gamma_j z_{ik} + \sum_{l=1}^{p-1} \delta_l w_{il}$$

  - The β's jointly quantify the effect of the design factor
  - The γ's jointly quantify the effect of nuisance factor 1
  - The δ's jointly quantify the effect of nuisance factor 2

- Three relevant hypotheses are:

$$H_0\text{: } \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0 \text{ vs. } H_A\text{: } \beta_j \neq 0 \text{ for some } j$$
$$H_0\text{: } \gamma_1 = \gamma_2 = \cdots = \gamma_{p-1} = 0 \text{ vs. } H_A\text{: } \gamma_k \neq 0 \text{ for some } k$$
$$H_0\text{: } \delta_1 = \delta_2 = \cdots = \delta_{p-1} = 0 \text{ vs. } H_A\text{: } \delta_l \neq 0 \text{ for some } l$$

- These hypotheses are tested by comparing a *full* model and *reduced* models
  - We try to determine whether the full model fits the data significantly better than the reduced one

LSDs to Compare Means

- Here we're interested in testing the following hypothesis (while accounting for the influence of the nuisance factors):

$$H_0\text{: } \mu_1 = \mu_2 = \cdots = \mu_p \text{ vs. } H_A\text{: } \mu_j \neq \mu_{j'} \text{ for some } j' \neq j$$

- We do this by testing

$$H_0\text{: } \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0 \text{ vs. } H_A\text{: } \beta_j \neq 0 \text{ for some } j$$

with an ANOVA in the context of the following linear regression model

$$Y_i = \alpha + \sum_{j=1}^{p-1} \beta_j x_{ij} + \sum_{k=1}^{p-1} \gamma_k z_{ik} + \sum_{l=1}^{p-1} \delta_l w_{il} + \varepsilon_i$$

- The relevant sums of squares are:
  - $$SS_T = \sum_{(j,k,l)\in\mathcal{S}} \sum_{i=1}^{n} (y_{ijkl} - \overline{y}_{....})^2$$
  - $$SS_C = \sum_{(j,k,l)\in\mathcal{S}} \sum_{i=1}^{n} (\overline{y}_{.j..} - \overline{y}_{....})^2 = np \sum_{j=1}^{p} (\overline{y}_{.j..} - \overline{y}_{....})^2$$
  - $$SS_{B_1} = \sum_{(j,k,l)\in\mathcal{S}} \sum_{i=1}^{n} (\overline{y}_{..k.} - \overline{y}_{....})^2 = np \sum_{k=1}^{p} (\overline{y}_{..k.} - \overline{y}_{....})^2$$
  - $$SS_{B_2} = \sum_{(j,k,l)\in\mathcal{S}} \sum_{i=1}^{n} (\overline{y}_{...l} - \overline{y}_{....})^2 = np \sum_{l=1}^{p} (\overline{y}_{...l} - \overline{y}_{....})^2$$

–

$$SS_E = \sum_{(j,k,l)\in\mathcal{S}} \sum_{i=1}^{n} \left(y_{ijkl} - \overline{y}_{\cdot j\cdot\cdot} - \overline{y}_{\cdot\cdot k\cdot} - \overline{y}_{\cdot\cdot\cdot l} + 2\overline{y}_{\cdot\cdot\cdot\cdot}\right)^2$$

- And the corresponding ANOVA table is shown below:

Source	SS	df	MS	Test Stat.
Design Factor	SS_C	$p-1$	$MS_C = \frac{SS_C}{p-1}$	$\frac{MS_C}{MS_E}$
Nuisance Factor 1	SS_{B_1}	$p-1$	$MS_{B_1} = \frac{SS_{B_1}}{p-1}$	$\frac{MS_{B_1}}{MS_E}$
Nuisance Factor 2	SS_{B_2}	$p-1$	$MS_{B_2} = \frac{SS_{B_2}}{p-1}$	$\frac{MS_{B_2}}{MS_E}$
Error	SS_E	$N-3p+2$	$MS_E = \frac{SS_E}{N-3p+2}$	
Total	SS_T	$N-1$		

- So how do we use this table?
  - We test

    $$H_0\colon\ \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0$$

    using $t_C \equiv \frac{MS_C}{MS_E}$
  - The p-value is: $\Pr(F_{p-1,\,N-3p+2} > t_C)$
  - We test

    $$H_0\colon\ \gamma_1 = \gamma_2 = \cdots = \gamma_{p-1} = 0$$

    using $t_{B_1} \equiv \frac{MS_{B_1}}{MS_E} \sim F_{(p-1,\,N-3p+2)}$
  - The p-value is:
  - We test

    $$H_0\colon\ \delta_1 = \delta_2 = \cdots = \delta_{p-1} = 0$$

    using $t_{B_2} \equiv \frac{MS_{B_2}}{MS_E} \sim F_{(p-1,\,N-3p+2)}$
  - The p-value is:

Netflix Example

Consider the latency experiment described at the beginning of Week 6 in which Netflix is experimenting with server-side modifications to improve (reduce) the latency of netflix.com. In particular, they have four different experimental conditions (A,B,C,D) that are intended to reduce average latency (in milliseconds). Two nuisance factors that may also influence latency are browser (Google Chrome, Microsoft Edge, Firefox, Safari), and time of day (00:01-06:00, 06:01-12:00, 12:01-18:00, 18:01-00:00). The design of the experiment is the 4×4 Latin square shown below. In order to determine whether the expected latency in each condition differs significantly, $n = 500$ users are randomized to each of the $p^2 = 16$ blocks.

		Browser			
		Chrome	Edge	Firefox	Safari
Time	00:01-06:00	A	B	C	D
	06:01-12:00	D	A	B	C
	12:01-18:00	C	D	A	B
	18:01-00:00	B	C	D	A

The data is analyzed with the following linear regression model

$$Y_i = \alpha + \beta_2 x_{i2} + \beta_3 x_{i3} + \beta_4 x_{i4} + \gamma_1 z_{i1} + \gamma_2 z_{i2} + \gamma_3 z_{i3} + \delta_2 w_{i2} + \delta_3 w_{i3} + \delta_4 w_{i4} + \varepsilon_i$$

where x_{i2}, x_{i3}, x_{i4} are indicators for conditions B,C,D (condition A is the baseline), z_{i1}, z_{i2}, z_{i3} are browser indicators for Microsoft Edge, Firefox and Safari (Google Chrome is the baseline), and w_{i2}, w_{i3}, w_{i4} are time indicators for time periods 2-4 (time period 1 is the baseline). The ANOVA table associated with this model is shown below.

Source	SS	df	MS	Test Stat.
Condition	203903.38	3	67967.79	679.14
Browser	32.95	3	10.98	0.1097
Time	333242.01	3	111080.67	1109.92
Error	799636.18	7990	100.08	
Total	1336815	7999		

LSDs to Compare Proportions

- Here we're interested in testing the following hypothesis (while accounting for the influence of the nuisance factors):

$$H_0\text{: } \pi_1 = \pi_2 = \cdots = \pi_p \text{ vs. } H_A\text{: } \pi_j \neq \pi_{j'} \text{ for some } j' \neq j$$

- We do this by testing

$$H_0\text{: } \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0 \text{ vs. } H_A\text{: } \beta_j \neq 0 \text{ for some } j$$

with a likelihood ratio test (LRT) in the context of the following logistic regression model

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \alpha + \sum_{j=1}^{p-1}\beta_j x_{ij} + \sum_{k=1}^{p-1}\gamma_k z_{ik} + \sum_{l=1}^{p-1}\delta_l w_{il}$$

  - The likelihood ratio test compares the full model to the one with out the x's

- Similarly, we test

$$H_0\text{: } \gamma_1 = \gamma_2 = \cdots = \gamma_{p-1} = 0 \text{ vs. } H_A\text{: } \gamma_k \neq 0 \text{ for some } k$$

with a LRT that compares the full model to the reduced one without the z's

- And we test

$$H_0\text{: } \delta_1 = \delta_2 = \cdots = \delta_{p-1} = 0 \text{ vs. } H_A\text{: } \delta_l \neq 0 \text{ for some } l$$

with a LRT that compares the full model to the reduced one without the w's

- The observed test statistic for all of these tests is

$$t = 2 \times \log\left(\frac{\text{Likelihood}_{\text{Full Model}}}{\text{Likelihood}_{\text{Reduced Model}}}\right)$$

$$= 2 \times [\text{Log-Likelihood}_{\text{Full Model}} - \text{Log-Likelihood}_{\text{Reduced Model}}]$$

difference in # of params
$= [3(p-1)+1] - (p) = 2p-2$

  - The p-value for all of these tests is:

$\Pr(\chi^2_{(2p-2)} > t)$

Uber Example

Consider an experiment in which Uber is investigating the influence of three different promotional offers on ride-booking-rate (RBR).

- Promo A: None
- Promo B: One free ride today
- Promo C: Book a ride today and get 50% off your next 2 rides

The experimenters would like to control for a possible day-of-week effect and so they want to block by day. They would also like to control for possible city-to-city differences and so they also want to block by city. To do so they run a 3×3 Latin square design as illustrated in Table 1. Interest lies in determining whether or not the different promotions perform similarly with respect to RBR – and they wish to determine which one maximizes RBR – while controlling for the effects of day and city. In order to do this they randomize $n = 1000$ users to each of the $p^2 = 9$ blocks.

Table 1: 3×3 Latin square design for the Uber experiment

		City		
		Toronto	Vancouver	Montreal
Day	Friday	A	B	C
	Saturday	C	A	B
	Sunday	B	C	A

The data is analyzed with the following logistic regression model

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \alpha + \beta_2 x_{i2} + \beta_3 x_{i3} + \gamma_1 z_{i1} + \gamma_2 z_{i2} + \delta_1 w_{i1} + \delta_2 w_{i2}$$

where x_{i2} and x_{i3} are condition indicators for promos B and C (promo A is the baseline), z_{i1} and z_{i2} are day indicators for Saturday and Sunday (Friday is the baseline), and w_{i1} and w_{i2} are city indicators for Toronto and Vancouver (Montreal is the baseline)

Optional Exercises:

- R Analysis: 19, 24