15.1 Examples of Within-Subjects Designs

- The simplest within-subjects (repeated measures) design involves a single independent variable, which Keppel symbolizes as (A x S). As a transition from the two-factor independent groups design, this notation emphasizes the idea that we treat participants as a factor in the repeated measures design.
- We can combine two within-subjects independent variables to produce a completely within-subjects design (A x B x S). In such a design, each participant will be exposed to each level of each variable.
- We can combine a within-subjects independent variable with a between-subjects independent variable to produce a mixed design (also called a split-plot design). Keppel indicates that a variable is between subjects by placing it outside the parentheses, so an A (B x S) would be a mixed design with A as a between-subjects factor and B as a within-subjects factor. In this design, a participant would be exposed to only one level of Factor A, but would be exposed to all levels of Factor B.
- “The main advantage of within-subjects designs is the control of participant variability — that is, individual differences.” The scores for a single participant will still vary due to random factors (as well as potential treatment effects), but we can expect that individual differences will not affect our MS_{Treatment} because the same person is responding to each level of treatment. Our MS_{Error} will ordinarily be smaller because it will reflect only random effects — with individual differences removed. Thus, the repeated measures design is more powerful than a comparable independent groups design. Because each individual is producing more than one piece of data, the repeated measures design is usually more efficient as well.
- However, the repeated measures design introduces the possibility of carryover effects, such as practice effects or fatigue effects. To control for such effects, it is necessary to counterbalance the order of treatments in a repeated measures design.
- We also have to be concerned about differential carryover effects in a repeated measures design. Counterbalancing will not address this particular problem.

15.2 Controlling Practice Effects

- Generally, if you have a sufficiently small number of conditions, you would use complete counterbalancing. A completely counterbalanced experiment is one in which every possible ordering of the conditions is used in the experiment an equal number of times. For you combinations/permutations freaks, that would be N! combinations (where N is the number of conditions). Thus, with 2 conditions there would be only 2 unique orders (AB and BA). With 3 conditions, there would be 6 unique orders (ABC, ACB, BAC, BCA, CAB, CBA). You would have to run an equal number of participants in each order, so a completely counterbalanced repeated measures design for an experiment with 4 conditions would involve a minimum of 24 participants and could also involve 48, 72, 96, participants, etc. You could not have a completely counterbalanced repeated measures design for an experiment with 4 conditions and have only 32 participants...because some orders would be over-represented.
- How many participants would you need to completely counterbalance a repeated measures design with 5 conditions? With 6 conditions? With 7 conditions? (You would need 120, 720, and 5040 participants, respectively!) So suppose that you are not inclined to run so many participants. Is there some way around this problem? Well, you could use an incomplete counterbalancing procedure.
- There are several different incomplete counterbalancing schemes, some of which are better than others. A simple Latin Square design is easy, but not particularly effective. For instance, suppose that you have 5 conditions (A, B, C, D, E). You could construct 5 unique orders that would insure that each condition occurred equally often in each of the 5 positions, as seen below.

<table>
<thead>
<tr>
<th></th>
<th>First</th>
<th>Second</th>
<th>Third</th>
<th>Fourth</th>
<th>Fifth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order 1</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>Order 2</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>A</td>
</tr>
<tr>
<td>Order 3</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>Order 4</td>
<td>D</td>
<td>E</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>Order 5</td>
<td>E</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
</tr>
</tbody>
</table>
Although this approach has some merit, including simplicity, it has the problem that the same conditions always follow one another. So even though it’s true that condition A will occur in each of the 5 positions, it always follows E (when it follows any condition) and it always precedes B. So if A produces particularly strong carry over effects, those effects tend to fall unevenly on condition B (and are not spread out over the other conditions).

• A better approach, though a bit more complicated, is illustrated below. This is a balanced Latin Square design and you should see that it is substantially better than the simple Latin Square illustrated above. The major advantage is that it not only insures that each condition occurs equally often in each position, but it also insures that different conditions precede and follow one another. Keppel calls this sort of incomplete counterbalancing digram-balanced.

### Figure 10–1
Illustration of algorithm to produce an incomplete counterbalancing scheme for $N$ conditions. The results next to the algorithm are for $N = 7$, with labels for the conditions: A B C D E F G.

- **15.3 Differential Carryover Effects**

  • These differential carryover effects arise when earlier treatments continue to exert an influence on subsequent responses, producing a treatment x position interaction. Generally, you can reduce the possibility of such effects by allowing sufficient time between treatments. If these effects cannot be removed, then you should probably consider using an independent groups design.