Regression analysis is widely used in applications, with which one quantifies the dependence of responses (outcomes, dependent variables) on a set of predictors/covariates (inputs, independent variables).

Data for regression analysis come from observational studies or designed experiments. With the former, one does not have much control but just records what happens. With the latter, one sets inputs to specified values then observes the resulting outcomes.

In some applications, the sole purpose is to predict the outcomes from the inputs, but in some others, one is more interested in how the inputs affect the outcomes. Designed experiments are primarily used to serve the latter purpose.

The input variables could be categorical or numerical by nature, but in designed experiments, numerical inputs are typically fixed to a few "grid" values (levels), and are often treated as categorical variables in analysis.

The outcomes are generally affected by many factors, and one hopes to account for as much a portion as possible via the available input variables; the portion unaccounted for are deemed as "errors," explicit with continuous responses and implicit with binary, ordinal, or count responses. Do include terms that may not be of direct interest, or else those would be shoveled into the "error" term.

Sometimes, one can control the inputs of interest but study subjects may carry extraneous variables, traits that are beyond control but could impact the outcomes systematically.

1 Factorial Design

With crossed factors, (full) factorial design is convenient if feasible. Consider factor A with a levels and factor B with b levels, the mean responses for the ab treatment combinations (cells) are generally different,

$$Y_{ijk} = \mu_{ij} + \epsilon_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk}, \tag{1}$$

$i = 1, \ldots, a$, $j = 1, \ldots, b$, $k = 1, \ldots, n_{ij}$. The design is balanced when $n_{ij} = n$ are all equal, which we assume hereafter.

Expressing the ab independent μ_{ij}'s by $\mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}$ is overparameterization, and to keep things identifiable, one needs a set of side conditions such as $\sum_i \alpha_i = 0$, $\sum_j \beta_j = 0$, and $\sum_i (\alpha\beta)_{ij} = \sum_j (\alpha\beta)_{ij} = 0$, for a total of $1 + 1 + (a + b - 1) = a + b + 1$ constraints. The degrees of freedom associated with α_i's is $(a - 1)$, with β_j's $(b - 1)$, and with $(\alpha\beta)_{ij}$'s $ab - (a + b - 1) = (a - 1)(b - 1)$.

Decomposing $Y_{ijk} - \bar{Y}_{\cdots} = (Y_{ijk} - \bar{Y}_{ij\cdot}) + (\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdot j\cdot} + \bar{Y}_{\cdots}) + (\bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdots}) + (\bar{Y}_{\cdot j\cdot} - \bar{Y}_{\cdots})$, one obtains the sum of squares associated with α_i, β_j, $(\alpha\beta)_{ij}$, and ϵ_{ijk} in

$$\begin{aligned}
\text{SSA} &= \textstyle\sum_{i,j,k} (\bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdots})^2 = \sum_{i,j,k} (\alpha_i + \bar{\epsilon}_{i\cdot\cdot} - \bar{\epsilon}_{\cdots})^2, \\
\text{SSB} &= \textstyle\sum_{i,j,k} (\bar{Y}_{\cdot j\cdot} - \bar{Y}_{\cdots})^2 = \sum_{i,j,k} (\beta_j + \bar{\epsilon}_{\cdot j\cdot} - \bar{\epsilon}_{\cdots})^2, \\
\text{SSAB} &= \textstyle\sum_{i,j,k} (\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdot j\cdot} + \bar{Y}_{\cdots})^2 = \sum_{i,j,k} ((\alpha\beta)_{ij} + \bar{\epsilon}_{ij\cdot} - \bar{\epsilon}_{i\cdot\cdot} - \bar{\epsilon}_{\cdot j\cdot} + \bar{\epsilon}_{\cdots})^2, \\
\text{SSE} &= \sum_{i,j,k} (Y_{ijk} - \bar{Y}_{ij\cdot})^2 = \sum_{i,j,k} (\epsilon_{ijk} - \bar{\epsilon}_{ij\cdot})^2,
\end{aligned}$$

where SSE has df $ab(n-1)$.

When $\sum_{i,j}(\alpha\beta)_{ij}^2$ is small (insignificant?), one often chooses to work with an additive model by setting $(\alpha\beta)_{ij}=0$, effectively shovling those into ϵ_{ijk}; $\text{SSE}^* = \text{SSAB} + \text{SSE}$ is the SSE for an additive model, with df $(abn-a-b+1)$. Additive models are easier to interpret, and make the $a \times b$ structure "meaningful."

When $(\alpha\beta)_{ij}$'s are not negligible, the decomposition $\mu_{ij} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}$ does not necessarily bring much benefit, and one could be better off treating things as one composite factor with ab levels.

When $n=1$, $(\alpha\beta)_{ij}$ and ϵ_{ij} are not identifiable from each other, so an additive model is typically the only viable solution. Nevertheless, John Tukey considered a one df interaction term of form $(\alpha\beta)_{ij} = \delta\alpha_i\beta_j$, and derived a test for $\delta = 0$.

An **experimental unit** (EU) is material to which a treatment is applied in a single trial of an experiment, and a **measurement unit** (MU) is material that is measured in an experiment. In the design above, the EUs coincide with the MUs, totaling abn units.

For the design to be effective uncovering discrepancies among the μ_{ij}'s, the magnitude of ϵ_{ijk} should be small and uniform, and the replication reduces the error magnitude in $\bar{Y}_{ij\cdot}$ by a factor of $1/\sqrt{n}$.

The EU's are assumed behaving "uniformly" for the purpose (no hidden extraneous traits), and should be assigned randomly to the treatment combinations (equal chance to any of the ab cells, or *completely randomized design*, CRD).

2 Blocking

Blocks are some physical entities on which experiments are conducted, and an additive block effect is typically a nuisance. Blocks crossed with treatment levels often help to enhance statistical power, whereas blocks nested under treatment levels are typically the experimental units.

2.1 Crossed Blocks

Materials (subjects) used in experiments often carry extraneous traits, of which the effects could be difficult or impossible to account for. *Assuming* the effects of the extraneous traits (subject effects) are additive to the treatment effects, one may use the "same" subjects multiple times, assigning a set of different treatment combinations in the process; the subject effects typically cancel out for the purpose of assessing the treatment effects.

Examples include crossover studies using human subjects (with washout periods between trials), before-after studies, agriculture experiments with treatments applied to subplots of homogeneous fields, etc.

Write $Y_{ij} = \mu+\alpha_i+\beta_j+\epsilon_{ij}$, $i=1,\ldots,a$, $j=1,\ldots,b$, where α_i's are the treatment effects satisfying $\sum_i \alpha_i = 0$ and β_j's are the nuisance subject (block) effects. This is a *complete block design* (CBD). When the assignment of treatments to the EUs in each block is randomized, one has a *randomized complete block design* (RCBD).

The sum of squares are as in §1 but with $n=1$ and $(\alpha\beta)_{ij}=0$; SSE there vanishes and SSAB there is the SSE here. SSB is usually not needed in the current setting.

Treatment effects are usually assessed via contrasts of α_i's, $\theta = \sum_i c_i\alpha_i$ for some known c_i's satisfying $\sum_i c_i = 0$, and $\tilde{Y}_i = \sum_i c_i Y_{ij} = \theta + \sum_i c_i\epsilon_{ij} = \theta + \tilde{\epsilon}_i$.

When the treatment has only two levels $i = 1, 2$, one has paired data.

When α_i's represent more than one factors, they can be further decomposed similar to the μ_{ij}'s in §1.

2.2 Nested Random Blocks

Blocks generally do not carry any meaning and are not reproducible in the future, so block effects are typically random by nature. One however may treat block effects as fixed in settings where they cancel out, such as in §2.1.

For an example of nested blocks, consider agriculture experiments where treatments are applied to plots of field (EUs), but responses are measured on individual plants (MUs).

Write $Y_{ijk} = \mu_i + g_{j(i)} + \epsilon_{ijk}$, where $\epsilon_{ijk} \sim N(0, \sigma^2)$, $g_{j(i)} \sim N(0, \tau^2)$. We assume the balanced case, with $i = 1, \ldots, a$, $j = 1, \ldots, b$, $k = 1, \ldots, n$, which allows clean formulas.

Inferences concerning μ_i are based on $\bar{Y}_{i\cdot\cdot} \sim N\big(\mu_i, \frac{1}{b}(\tau^2 + \sigma^2/n)\big)$, and $\sum_{i,j}(\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot})^2 = \sum_{i,j}(g_{j(i)} - \bar{g}_{\cdot(i)} + \bar{\epsilon}_{ij\cdot} - \bar{\epsilon}_{i\cdot\cdot})^2$ has expectation $a(b-1)(\tau^2 + \sigma^2/n)$, so one may simply work with the block means $\bar{Y}_{ij\cdot} = \mu_i + g_{j(i)} + \bar{\epsilon}_{ij\cdot} = \mu_i + e_{ij}$; remember that the blocks are EUs. This works as long as the block size n is fixed, regardless whether $j(i)$'s are balanced.

2.3 Split-Plots

In §1, the abn EUs are evenly allocated to the ab cells of treatment combinations, and the index k has no meaning so can be arbitrarily permuted intra-cell.

Now suppose one can only assign the a levels of factor A to an blocks, but each block is divided into b subplots to receive the b levels of factor B. This leads to a split-plot design, with the an blocks as the EUs for factor A and the abn subplots as the EUs for factor B.

Write $Y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + g_{k(i)} + \epsilon_{ijk}$, where $\epsilon_{ijk} \sim N(0, \sigma^2)$, $g_{k(i)} \sim N(0, \tau^2)$. The index k is now meaningful, nested under levels of factor A but crossed with levels of factor B, and $\sum_{i,j,k}(Y_{ijk} - \bar{Y}_{ij\cdot})^2$ involves both $g_{k(i)}$ and ϵ_{ijk}. One has

$$\begin{aligned}\sum_{i,j,k}(Y_{ijk} - \bar{Y}_{ij\cdot})^2 &= \sum_{i,j,k}(\bar{Y}_{i\cdot k} - \bar{Y}_{i\cdot\cdot})^2 + \sum_{i,j,k}(Y_{ijk} - \bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot k} + \bar{Y}_{i\cdot\cdot})^2\\ &= \sum_{i,j,k}(g_{k(i)} - \bar{g}_{\cdot(i)} + \bar{\epsilon}_{i\cdot k} - \bar{\epsilon}_{i\cdot\cdot})^2 + \sum_{i,j,k}(\epsilon_{ijk} - \bar{\epsilon}_{ij\cdot} - \bar{\epsilon}_{i\cdot k} + \bar{\epsilon}_{i\cdot\cdot})^2\\ &= \text{SSBlk} + \text{SSE},\end{aligned}$$

where SSE has df $a(b-1)(n-1)$, SSBlk has df $a(n-1)$, and MSBlk = SSBlk/$a(n-1)$ has mean $\tau^2 + \sigma^2/b$. The sum of squares associated with α_i, β_j, and $(\alpha\beta)_{ij}$ are seen to be

$$\begin{aligned}\text{SSA} &= \textstyle\sum_{i,j,k}(\bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdots})^2 = \sum_{i,j,k}(\alpha_i + \bar{g}_{\cdot(i)} - \bar{g}_{\cdot(\cdot)} + \bar{\epsilon}_{i\cdot\cdot} - \bar{\epsilon}_{\cdots})^2,\\ \text{SSB} &= \textstyle\sum_{i,j,k}(\bar{Y}_{\cdot j\cdot} - \bar{Y}_{\cdots})^2 = \sum_{i,j,k}(\beta_j + \bar{\epsilon}_{\cdot j\cdot} - \bar{\epsilon}_{\cdots})^2,\\ \text{SSAB} &= \textstyle\sum_{i,j,k}(\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdot j\cdot} + \bar{Y}_{\cdots})^2 = \sum_{i,j,k}((\alpha\beta)_{ij} + \bar{\epsilon}_{ij\cdot} - \bar{\epsilon}_{i\cdot\cdot} - \bar{\epsilon}_{\cdot j\cdot} + \bar{\epsilon}_{\cdots})^2.\end{aligned}$$

Test for $\sum_{ij}(\alpha\beta)_{ij}^2 = 0$ is based on SSAB/SSE, and the SSE for an additive model is $\text{SSE}^* = \text{SSAB} + \text{SSE} = \sum_{i,j,k}(Y_{ijk} - \bar{Y}_{i\cdot k} - \bar{Y}_{\cdot j\cdot} + \bar{Y}_{\cdots})^2$, with df $(b-1)(an-1)$. In an

additive model, one may test for $\sum_i \alpha_i^2 = 0$ using SSA/SSBlk, and test for $\sum_j \beta_j^2 = 0$ using SSB/SSE*.

2.4 Split-Split-Plots

Now suppose each EU of factor B in §2.3 is further partitioned into c subsubplots to receive the levels of factor C. One may write $Y_{ijkl} = \mu_{ijk} + g_{l(i)} + h_{j,l(i)} + \epsilon_{ijkl}$, where $i = 1, \ldots, a$, $j = 1, \ldots, b$, $k = 1, \ldots, c$, $l = 1, \ldots, n$, for μ_{ijk} fixed and $g_{l(i)}$, $h_{j,l(i)}$, ϵ_{ijkl} random with variances σ_g^2, σ_h^2, σ^2; $g_{l(i)}$ is the block effect and $h_{j,l(i)}$ is the B $\times$ Blk interaction. One has

$$\begin{aligned}
\textstyle\sum_{i,j,k,l}(Y_{ijkl} - \bar{Y}_{ijk\cdot})^2 =& \textstyle\sum_{i,j,k,l}(\bar{Y}_{i\cdot\cdot l} - \bar{Y}_{i\cdots})^2 + \sum_{i,j,k,l}(Y_{ij\cdot l} - \bar{Y}_{ij\cdot\cdot} - \bar{Y}_{i\cdot\cdot l} + \bar{Y}_{i\cdots})^2 \\
&+ \sum_{i,j,k,l}(Y_{ijkl} - \bar{Y}_{ijk\cdot} - \bar{Y}_{ij\cdot l} + \bar{Y}_{ij\cdot\cdot})^2 \\
=& \textstyle\sum_{i,j,k,l}(g_{l(i)} - \bar{g}_{\cdot(i)} + \bar{h}_{\cdot l(i)} - \bar{h}_{\cdot\cdot(i)} + \bar{\epsilon}_{i\cdot\cdot l} - \bar{\epsilon}_{i\cdots})^2 \\
&+ \textstyle\sum_{i,j,k,l}(h_{jl(i)} - \bar{h}_{j\cdot(i)} - \bar{h}_{\cdot l(i)} + \bar{h}_{\cdot\cdot(i)} + \bar{\epsilon}_{ij\cdot l} - \bar{\epsilon}_{ij\cdot\cdot} - \bar{\epsilon}_{i\cdot\cdot l} + \bar{\epsilon}_{i\cdots})^2 \\
&+ \sum_{i,j,k,l}(\epsilon_{ijkl} - \bar{\epsilon}_{ijk\cdot} - \bar{\epsilon}_{ij\cdot l} + \bar{\epsilon}_{ij\cdot\cdot})^2 \\
=& \text{SSBlk1} + \text{SSBlk2} + \text{SSE},
\end{aligned}$$

where SSE has df $ab(c-1)(n-1)$, SSBlk2 has df $a(b-1)(n-1)$ and MSBlk2 estimates $\sigma_h^2 + \sigma^2/c$, SSBlk1 has df $a(n-1)$ and MSBlk1 estimates $\sigma_g^2 + \sigma_h^2/b + \sigma^2/bc$.

As usual, SSTr $= \sum_{i,j,k,l}(\bar{Y}_{ijk\cdot} - \bar{Y}_{\cdots\cdot})^2$ can be decomposed into terms of main effects, two-way interactions, and a three-way interaction; SSA involves all random terms, SSB and SSAB involve $h_{j,l(i)}$ and ϵ_{ijkl} but not $g_{l(i)}$, and the remaining terms (all including factor C) only involve ϵ_{ijkl}.

One may test for the three-way interaction using MSABC/MSE. Absent the three-way interaction, one may form SSE* = SSE + SSABC, and two-way interactions could be tested using MSAB/MSBlk2, MSAC/MSE*, and MSBC/MSE*; when either or both of the latter two are insignificant, one may choose to eliminate the term(s) in the model and add more term(s) into SSE*. When interactions involving factor A are absent, one may test for the A main effect using MSA/MSBlk1. When interactions involving factor B are absent, one may test for the B main effect using MSB/MSBlk2* for SSBlk2* = SSBlk2 + SSAB, and when interactions involving factor C are absent, one may test for the C main effect using MSC/MSE*.

3 Nested Factors

With crossed factors, say A and B, any level of factor A can be combined with any level of factor B to form a treatment combination. This is the more common scenario but there are exceptions.

Consider a study of automobile fuel economy, where car models (factor B) are nested under car makes (factor A); under each level of factor A, there is an entirely different set of levels of factor B.

Write $Y_{ijk} = \mu + \alpha_i + \beta_{j(i)} + \epsilon_{ijk}$, $i = 1, \ldots, a$, $j = 1, \ldots, b_i$, $k = 1, \ldots, n_{ij}$, with side conditions $\sum_i (\sum_j n_{ij})\alpha_i = 0$ and $\sum_j n_{ij}\beta_{j(i)} = 0$, $\forall i$.

One may calculate SSA $= \sum_{i,j,k}(\bar{Y}_{i\cdot\cdot} - \bar{Y}_{\cdots})^2$ with df $(a-1)$, SSB(A) $= \sum_{i,j,k}(\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot})^2$ with df $\sum_i (b_i - 1)$, and SSE $= \sum_{i,j,k}(Y_{ijk} - \bar{Y}_{ij\cdot})^2$ with df $\sum_{i,j} b_j(n_{ij} - 1)$. It may not make much sense to desire "balanced" design in the setting, especially for index j, and the dot notation is a bit "abusive" as the averaging is generally over different number of observations at different "localities."

Aggregating i-specific $\sum_{j,k}(\bar{Y}_{ij\cdot} - \bar{Y}_{i\cdot\cdot})^2$ into SSB(A) may not be desirable, and one could be better off performing separate one-way analysis for each level of factor A.

3.1 A Toy Example

Consider the setting of §2.2, $Y_{ijk} = \mu + \alpha_i + g_{j(i)} + \epsilon_{ijk}$, where $\epsilon_{ijk} \sim N(0, \sigma^2)$, $g_{j(i)} \sim N(0, \tau^2)$. Setting $\tau^2 = \infty$, the nested random blocks become the levels of nested factor. Setting $\tau^2 = 0$, one ignores blocking and treats the MUs as EUs.

A "balanced" toy example is created with $a = 3$, $b = 3$, and $n = 4$, for a total of 36 entries of Y_{ijk}. `R` data frame `toy` has elements `y`, `trt`, and `blk`, where `blk` has values `1:9`. A "replicate" of `toy` is in `toy1`, where `blk` values `4:6` and `7:9` are coded as `1:3`. In data frame `toy2`, the 36 entries are collapsed down to 9 entries of block averages.

For `blk` as fixed effect nested under `trt`, `toy` and `toy1` are equivalent.

```
summary(aov(y~trt/blk,toy)); summary(aov(y~trt/blk,toy1))
summary(lm(y~trt/blk,toy)); summary(lm(y~trt/blk,toy1))
```

For `blk` random nested under `trt`, `toy1` is not usable, and the following are equivalent.

```
toy.fit=lmer(y~trt+(1|blk),toy)
anova(toy.fit); summary(toy.fit)
summary(aov(y~trt,toy2)); summary(lm(y~trt,toy2))
```

With unbalanced design where different EUs may contain different number of MUs, the `lmer` fit should be used.

Ignoring blocks, MUs are taken as EUs.

```
summary(aov(y~trt,toy)); summary(lm(y~trt,toy))
```

4 Examples

Example 1 *To study the effect of pesticides on bird, 65 chicks are randomly assigned to 5 diets, one control and 4 containing different pesticides.*

This is one-way CRD, a special case of the design in §1 with $b = 1$. SSA has df 4 and SSE has df 60; SSB and SSAB vanish. The chicks do not have to be evenly divided among the 5 treatment levels.

Example 2 *Four different salt-sand mixtures are tested on 4 road sections for winter road treatments. Each road section is divided into 4 portions to receive the 4 mixtures, randomly assigned.*

This is RCBD of §2.1. SSA has df 3 and SSE has df 9.

Example 3 *To study the effects of feral pigs on the native vegetation in Santa Cruz Island, a researcher collected data from* $10 \times 2 \times 2 = 40$ *plots,* $2 \times 2 = 4$ *plots each around 10 oak trees on the island; the two binary factors are under the canopy or not (factor C) and fenced or not (factor F).*

This is RCBD of §2.1, but with the 4 levels of treatment further decomposed into a 2×2 structure. The 10 oak trees are the blocks. The 3 df SSA in §2.1 is decomposed into SSC+SSF+SSCF, each with df 1; this is the same SSA+SSB+SSAB decomposition in §1.

Example 4 *In an experiment on the effect of treatments A and B on the amount of substance S in mice's blood, it was not practical to use more than 4 mice on any one day. The treatments formed a* 2×2 *system:* $(A_0, A_1) \times (B_0, B_1)$*. The mice used on one day were all of the same sex. The data are recorded in the following table.*

Sex	*Day*	A_0B_0	A_1B_0	A_0B_1	A_1B_1
Male	*1*	*4.4*	*2.8*	*4.8*	*6.8*
	2	*5.3*	*3.3*	*1.9*	*8.7*
	4	*1.8*	*2.6*	*3.1*	*4.8*
	7	*5.4*	*6.9*	*6.2*	*9.3*
Female	*3*	*5.3*	*7.0*	*4.3*	*7.2*
	5	*3.7*	*5.9*	*6.2*	*5.1*
	6	*6.5*	*5.4*	*5.7*	*6.7*
	8	*5.2*	*6.8*	*7.9*	*7.9*

This is the split-plot design of §2.3, with days as random blocks, nested under the sex effect and crossed with the treatment effect; the 4-level treatment is further decomposed into a 2×2 structure.

SSE has df 18, SSBlk has df 6, SSs has df 1, SSTr has df 3, and SSsTr has df 3. Absent the s $\times$ Tr interaction, SSE* has df 21.

The data are in an `R` data frame `mice` with elements `A`, `B`, `sex`, `day`, and `S`. One may fit models and check results after loading packages `lme4` and `lmerTest`.

```
fit0=lmer(S~sex*(A*B)+(1|day),data=mice); summary(fit0)
fit1=lmer(S~sex+(A*B)+(1|day),data=mice); summary(fit1)
```

5 Analysis of Paired Data

Setting $a = 2$ in §2.1, one gets paired data.

5.1 Paired t-Test

Working with $d_j = Y_{2j} - Y_{1j} = (\alpha_2 - \alpha_1) + (\epsilon_{2j} - \epsilon_{1j}) = 2\alpha_2 + e_j$, one may test for $\alpha_2 = 0$ using a one-sample t-test, known as paired t-test. Paired t-test is equivalent to the F-test based on SSA/SSE in §2.1.

5.2 Before-After Studies

Before-after studies are commonly used to assess the effect of intervention, in which the measure of interest (Y) is taken on each subject before and after some training/intervention session; characteristics of the subjects ($\boldsymbol{x}$) may also be collected.

For Y continuous, one may write $Y_{ij} = \mu + \alpha_i + f(\boldsymbol{x}_j) + g(i, \boldsymbol{x}_j) + \epsilon_{ij}$, where $i = 1, 2$ codes before-after and $j = 1, \ldots, n$ labels the subjects. Note that no model forms are specified for $f(\boldsymbol{x})$ and $g(i, \boldsymbol{x})$ here. For an additive model with $g(i, \boldsymbol{x}_j) = 0$, this reduces to the setting of §5.1 with $\beta_j = f(\boldsymbol{x}_j)$, and $\boldsymbol{x}_j$ are not needed for the assessment of the intervention effect α_i.

In general, the intervention effect may vary with $\boldsymbol{x}$, and for continuous Y, a common practice is to model $Y_{2j} - Y_{1j} = (\alpha_2 - \alpha_1) + (g(2, \boldsymbol{x}_j) - g(1, \boldsymbol{x}_j)) + (\epsilon_{2j} - \epsilon_{1,j})$. Assuming $g(2, \boldsymbol{x}) - g(1, \boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{\beta}$, say, one should expect the same inferential results concerning $\boldsymbol{\beta}$ using either `lm(y2-y1~x1+x2)` or `lm(y~id+trt+trt:(x1+x2))`, where `trt` represents α_i and `id` represents $\beta_j = f(\boldsymbol{x}_j)$. For Y not continuous, literal differencing `y2-y1` no longer makes sense, but the model formula `y~id+trt+trt:(x1+x2))` can be used to achieve the effect.

5.3 A Toy Example

Generating synthetic data.

```
x=rnorm(20); y1=rnorm(20); y2=y1+2*x+rnorm(20,,0.3)
dat=data.frame(trt=as.factor(rep(1:2,c(20,20))),
               id=as.factor(c(1:20,1:20)),y=c(y1,y2),x=c(x,x))
```

Paired t-test and equivalents; this makes no practical sense for the data generated, but the purpose here is to numerically verify the equivalence of different implementations.

```
t.test(y2-y1)  ## paired t-test
summary(lm(y~trt+id,dat))  ## RCBD with fixed blocks
summary(lmer(y~trt+(1|id),dat))  ## random blocks
```

Modeling $g(2, \boldsymbol{x}_j) - g(1, \boldsymbol{x}_j)$

```
summary(lm(y2-y1~x))  ## direct differencing
summary(lm(y~id+trt+trt:x,dat))  ## implicit differencing
```

5.4 Crippled Pairs

Suppose one expects to collect paired data, but some pairs have one arm missing; technically this could be viewed as unbalanced block design. Instead of dropping the crippled pairs, one may include all available data using `lmer(y~trt+(1|id))`.

When all pairs are crippled, β_j and ϵ_{ij} are not identifiable from each other, and one is left with independent samples for use in a two-sample t-test.

5.5 Another Toy Example

Generating synthetic data; `trt1` is unbalanced.

```
x=rnorm(20); y1=x+rnorm(x,,0.3); y2=x+rnorm(x,0.5,0.3)
dat=data.frame(trt1=as.factor(rep(c("a","b","a","b"),c(9,11,9,11))),
               trt2=as.factor(rep(1:2,c(20,20))),
               blk=as.factor(c(1:20,1:20)),y=c(y1,y2))
```

This is a split-plot design with the whole plots the EUs of `trt1` and the subplots the EUs of `trt2`; both factors are binary. In an additive model $Y_{ijk} = \mu + \alpha_i + \beta_j + g_{k(i)} + \epsilon_{ijk}$, the test for α_i (MSA/MSBlk) is the same as a two-sample t-test, and that for β_j (MSB/MSE*) is the same as a paired t-test; need package `lmerTest`.

```
fit=lmer(y~trt1+trt2+(1|blk),dat)  ## split-plot additive model
anova(fit)
t.test((y1+y2)[1:9],(y1+y2)[10:20],var.equal=TRUE)  ## t-test for trt1
t.test(y1,y2,paired=TRUE)  ## t-test for trt2
```

Now ignore `trt1`, test for the effect of `trt2` using partly paired data, and compare with results using only pairs.

```
fit=lmer(y~trt2+(1|blk),dat[5:34,])
anova(fit)
t.test(y1[5:14],y2[5:14],paired=TRUE)
summary(lm(y~trt2+blk,dat[c(5:14,25:34),]))
```

5.6 Difference of Differences

Suppose one is to compare two therapies using a crossover design, and the effect of a therapy on a patient is assessed via the difference between the measurements at the onset and the exit of the treatment duration. With continuous measurements, one may perform a one-sample t-test using difference of differences, a natural extension of the paired t-test.

A model reflecting the scenario is seen to be $Y_{ijk} = \mu_{ij} + \beta_{ik} + \beta_{jk} + \epsilon_{ijk}$, where $i = 1, 2$ denote the two therapies, $j = 1, 2$ mark onset/exit, $k = 1, \ldots, n$ label the subjects, and the β's represent inter-subject/period variability. One has $(Y_{22k} - Y_{21k}) - (Y_{12k} - Y_{11k}) = \mu_{22} - \mu_{21} - \mu_{12} + \mu_{11} + \epsilon_{22k} - \epsilon_{21k} - \epsilon_{12k} + \epsilon_{11k} = \delta + e_k$, where the (i, j)-interaction δ quantifies the difference in the treatment effects of the two therapies.

Generating synthetic data.

```
x=rnorm(20); xx=rnorm(20)
y1=rnorm(20); y2=y1+0.5*x+rnorm(x,1,0.3)
y3=rnorm(20); y4=y3+1.5*x+rnorm(x,1.5,0.3)
dat=data.frame(trt1=as.factor(rep(1:2,rep(40,2))),
               trt2=as.factor(rep(c(1:2,1:2),rep(20,4))),
               id=as.factor(rep(1:20,4)),
               y=c(y1,y2,y3,y4),x=rep(x,4),xx=rep(xx,4))
```

Equivalent implementations of one-sample t-test using difference of differences; the implicit version could be used for non-continuous responses.

```
t.test(y4-y3-y2+y1)  ## diff of diff
t.test(y4-y3,y2-y1,paired=TRUE)
summary(lm(y~(id+trt1+trt2)^2,dat))  ## implicit d-o-d in trt1:trt2
```

Regressing difference of differences on covariates.

```
summary(lm(y4-y3-y2+y1~x+xx))  ## direct differencing
summary(lm(y~(id+trt1+trt2)^2+trt1:trt2:(x+xx),dat))  ## implicit d-o-d
```