Script to produce and investigate the Farwell and Herzberg (2003) plaid design

For this design Viewing order and Raters are regarded as sources of variation and so Viewings is crossed with Raters

- The first-phase design is a systematic design;
- The second-phase design is a split-unit design, with the first-phase factors allocated using a plaid design.

Initialize

```
library(knitr)
#knitr::spin("FHdesign.r")
library(dae)
```

```
## Loading required package: ggplot2
```

```
## Need help? Enter help(package = 'dae') and click on 'User guides, package vignettes
## and other docs'.
```

```
packageVersion("dae")
```

```
## [1] '3.1.37'
```

```
options(width = 95)
```

Generate first-phase systematic design

```
ph1.sys <- cbind(fac.gen(list(Expressiveness = c("Exp", "Unexp"), Patients = 4, Occasions = 2)),
                 fac.gen(list(Motions = c("Act", "Pas")), times = 8))
```

Generate the two-phase systematic design

```
ph2.sys <- cbind(fac.gen(list(Raters = 74, Viewings = 16)),
                 fac.gen(list(Trainings = c("Co", "Tr"), 16), times = 37),
                 rep(ph1.sys, times =74))
```

Randomize the two-phase design

```
ph2.lay <- designRandomize(allocated = ph2.sys[c("Trainings", "Expressiveness", "Patients",
                                                 "Occasions", "Motions")],
                           recipient = ph2.sys[c("Raters", "Viewings")],
                           except = "Viewings",
                           seed = 15674)
```

Plot the design

```
ph2.lay$Videos <- with(ph2.lay, fac.combine(list(Motions, Expressiveness, Patients)))
levels(ph2.lay$Videos) <- as.vector(t(outer(c("A,E,", "A,U,", "P,E,", "P,U,"),
                                            as.character(1:4), paste0)))
designGGPlot(ph2.lay, row.factors = "Raters", column.factors = "Viewings",
             labels = "Videos", label.size = 3,
             cellfillcolour.column = "Trainings", cellalpha = 0.5,
```

```
            title = NULL, title.size = 12, axis.text.size = 10,
            ggplotFuncs = list(theme(legend.position = "bottom")))
```

Convert the names of the factors to single capital letters

```
ph2.L.lay <- ph2.lay
names(ph2.L.lay)[match(c("Raters", "Viewings", "Trainings", "Expressiveness", "Patients",
                         "Occasions", "Motions"), names(ph2.L.lay))] <- c("R", "V", "T",
                                                                          "E", "P", "O", "M")
```

Produce the anatomy of the design for the homogeneous allocation model without random intertier interactions

- The design is orthogonal because all aefficiency values are 1.

```
ph2.canon <- designAnatomy(formulae = list(episo = ~ R * V,
                                           video = ~ (E/P)*O,
                                           alloc = ~ T * M * E),
                           data = ph2.L.lay)
summary(ph2.canon, which.criteria = "aeff")
```

```
##
##
## Summary table of the decomposition for episo, video & alloc
##
##  Source.episo df1  Source.video df2  Source.alloc df3  aefficiency
##  R              73                   T               1      1.0000
##                                      Residual       72
##  V              15 E                1 E               1      1.0000
##                    P[E]             6                          1.0000
##                    O                1 M               1      1.0000
##                    E#O              1 M#E             1      1.0000
##                    P#O[E]           6                          1.0000
##  R#V          1095                    T#M             1      1.0000
##                                       T#E             1      1.0000
##                                       T#M#E           1      1.0000
##                                       Residual     1092
```

Produce the anatomy for the homogeneous allocation model with all intertier interactions

- The design is orthogonal because all aefficiency values are 1.

```
ph2.homogint.canon <- designAnatomy(formulae = list(episo  = ~ R * V,
                                                    train  = ~ T * V,
                                                    video  = ~ R * O * (E/P),
                                                    motion = ~ T * M * (E/P) + M * R),
                                    data = ph2.L.lay)
summary(ph2.homogint.canon, which.criteria = "aeff")
```

```
##
##
## Summary table of the decomposition for episo, train, video & motion
##
```

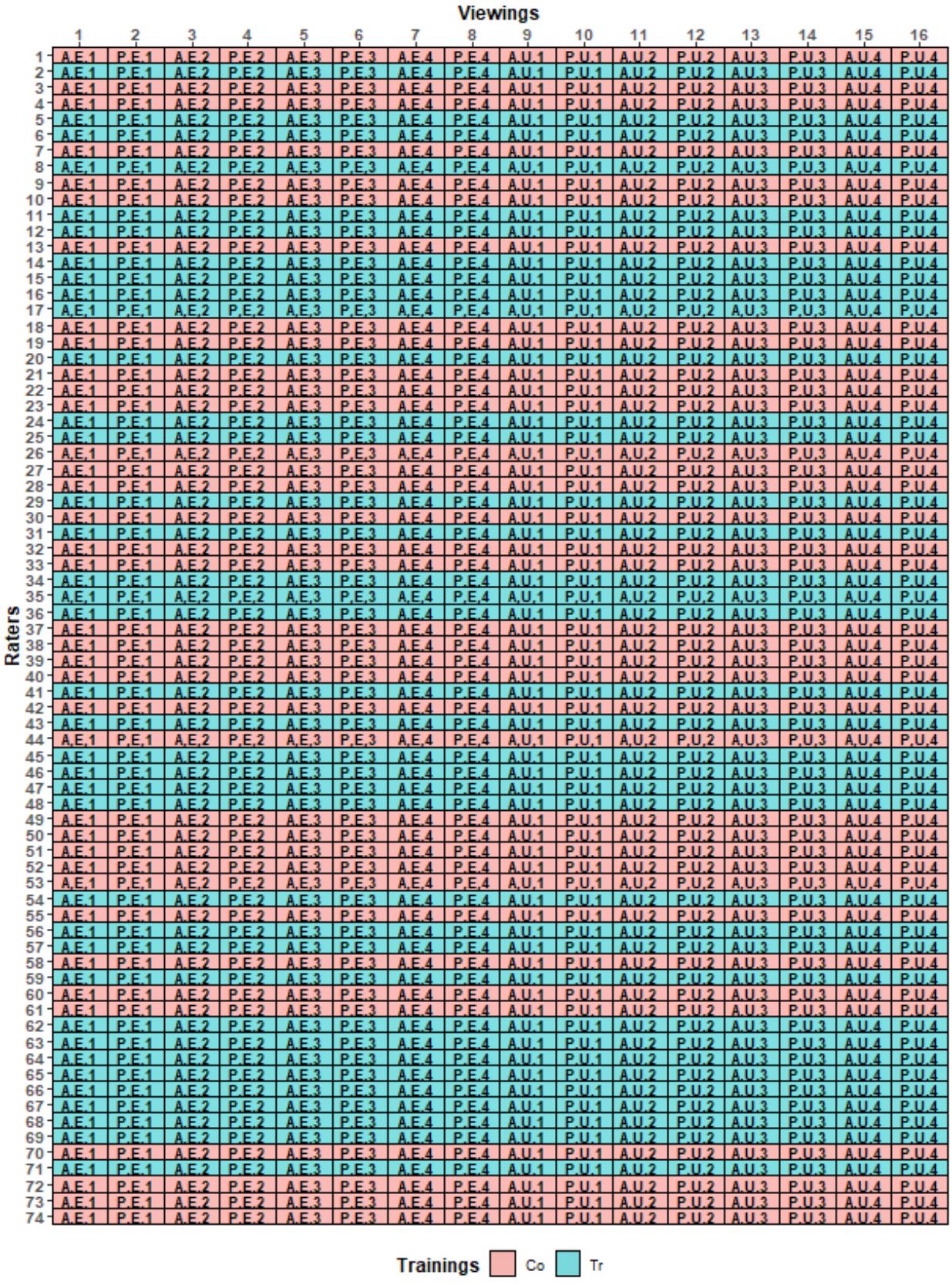

Figure 1: Farewell and Herzberg design that uses a plaid design. It shows the allocation videos to Viewings, with the colour corresponding to Trainings

```
##  Source.episo df1  Source.train df2  Source.video df3  Source.motion df4  aefficiency
##  R               73 T                1 R                1 T                 1      1.0000
##                     Residual        72 R               72 R[T]             72      1.0000
##  V               15 V                15 O                1 M                 1      1.0000
##                                         E                1 E                 1      1.0000
##                                         P[E]             6 P[E]              6      1.0000
##                                         O#E              1 M#E               1      1.0000
##                                         O#P[E]           6 M#P[E]            6      1.0000
##  R#V           1095 T#V              15 R#O              1 T#M               1      1.0000
##                                         R#E              1 T#E               1      1.0000
##                                         R#P[E]           6 T#P[E]            6      1.0000
##                                         R#O#E            1 T#M#E             1      1.0000
##                                         R#O#P[E]         6 T#M#P[E]          6      1.0000
##                     Residual      1080 R#O             72 M#R[T]           72      1.0000
##                                         R#E             72                          1.0000
##                                         R#P[E]         432                          1.0000
##                                         R#O#E           72                          1.0000
##                                         R#O#P[E]       432                          1.0000
```

Save the image

```
save.image("FHdesign.RData")
```