**R functions for obtaining the effective degrees of freedom**

An R function getVcEDF() was developed for improving the estimates of the variance components and generating the effective degrees of freedom (EDF). This function improves the estimation of the variance components by using the restricted maximum likelihood (REML) technique. The EDF is used to assess the effectiveness of the improved estimates.

The function getVcEDF()consists of three steps. The first step constructs the Fisher’s information matrix using the MS and DF extracted from an ANOVA table. The second step obtains the G matrix, which is used for changing the variables of interest. The last step is to improve the variance component estimates and to compute the EDF.

As mentioned, the first part of the getVcEDF() constructs the Fisher’s information matrix. In order to do this, it is necessary to extract the MS and DF from an ANOVA table. Normally, the basic R function aov() can generate the ANOVA table, but it only implements a single stage of decomposition, thus this cannot be applied directly to two-phase experiments. Two-phase experiments require two stages of decomposition: decomposition of the information from the Phase 1 block structure in the Phase 2 bock structure, and decomposition the information from the treatment structure in the Phase 1 block structure. Based on this idea of two stages of decomposition the aov() function can be applied twice, i.e. once for each stage of decomposition. The summary.aov.twoPhase() function from the infoDecompuTE package which implements the two stages of decomposition as described above, is therefore used to generate the ANOVA table.

The Fisher’s information matrix is then constructed by extracting the MS and DF from the ANOVA table. The appendix shows mathematically that the Fisher’s information matrix is equal to the DF divided by the twice of the square of the MS.

The sources of variation in the ANOVA table can be either fixed or random. The MS and DF are extracted from the sources of variation should not contain any fixed effect. This is because the variances should only be estimated from the sources of variation containing the random effects. However, there are some cases where the fixed effects are confounded with the random effects, e.g. balanced incomplete block designs. In these cases, the amount of confounding treatment information can be small enough to be neglected. This issue is out of scope for this write-up and therefore will not be addressed.

The second step is to construct the G matrix. The G matrix is extracted from the ANOVA table generated from the summary.aov.twoPhase() function. Note that the score function and the expected Fisher’s information matrix are with respect to the MS, but what we want to estimate are the individual variance components which make up the EMS. Hence, we want to transform the score function and the expected Fisher’s information matrix with respect to MS, to with respect to the variance component estimates, denoted by a vector . This transformation can be achieved by using the m × k G matrix, if there are k variance components to be estimated, where each element of the G matrix is. Hence, the expected mean squares can also be written as . This technique is also known as *change of variables*.

The G matrix that is extracted here is different to Jarrett and Ruggiero (2008). To enable the R function that to be used for every experimental design, the G matrix that is generated here also contains the coefficients of the variance components. This is different to Jarrett and Ruggiero (2008), where they used a binary matrix. Having the coefficients in the G matrix, it allows parameters of interest, a vector , to contain the individual variance components and each variance component has coefficient of one. This G matrix is used because sometimes the structure and the coefficients of the variance components of the EMS are not always what we expect for different sources of variation in the ANOVA table. Hence, by using this type of G matrix, it avoids the need to study every theoretical ANOVA table and adjust these coefficients with different linear combination of the variance component for a complicated analysis.

Note the first step only extracts the MS and DF of the source of variation without the treatment information. Hence, the G matrix extracted in this step has to match the sources of variation that were extracted in the previous step.

The third step is to estimate and optimise the variance components and compute the EDF. The variance components can be estimated based on the linear combination of the EMS and the experimental mean squares based on given data. However, the estimation of the variance components can be further improved using the REML which requires the construction of the Fisher’s information matrix and score function. The EDF can then be approximated as twice the square of the mean divided by the variance.

We will show the mathematical procedure on improving the variance component estimates using the REML technique. Note the expected Fisher’s information matrix with respect to the expected mean squares, , can be written as

.

Since what we are interested in is the variance components, , the expected Fisher’s information matrix with respect to , denoted by , can be generated from pre- and post-multiplying the by the G matrix, i.e.

The score function with respect to is obtained by multiplying the transpose of the G matrix by the first derivative of the likelihood function, this can be written as

From this, the Fisher’s scoring algorithm in REML, which is an iterative method for estimating the optimised variance components, , can be derived by

.

The Fisher’s information matrix and score function are iteratively updated using the newly optimised variance components . Note that the expected mean squares, , are also continuously updated as

.

This iterative algorithm will stop when has converged.

The formula for computing the EDF from Richard and Kathy (2008) calculated as twice the square of the mean divided by the variance. In order to calculate the EDF, it is necessary to know the variances of the parameters of interest. The variances can be obtained by calculating the sum of the elements of interest from the variance-covariance matrix. The variance covariance matrix is generated from the inverse of the Fisher’s information matrix. However, since the variance components in have coefficients of one, these coefficients have to be re-adjusted based on the variance components structure from the ANOVA table. This adjustment is based on the idea for calculating the variance of the linear combination of two random variables X and Y, i.e.

.

The end result of this step and the function getVcEDF() is the EDF for every source of variation without the treatment information and the newly optimised variance components.

In addition, two further R functions were developed to aid our study for different experimental designs. Since we cannot determine the variance component estimates and EDF based on one set of data, simData() function is used to generate the simulated datasets. The simData() function then computes the variance component estimates and EDF which incorporates the summary.aov.twoPhase() and getVcEDF() functions. Another function plotEDF() was developed which generates the line plots of EDF under different combinations of the variance components’ ratios. The examples of the EDF plots are presented in later part of this write-up.

**Example**

Consider a completely randomised design with 8 animals and 2 treatments for the first phase experiment, and proteomics experiment using iTRAQ labelled peptides for the second phase experiment. In particular, consider the situation in which there are 4 proteomic samples simultaneously assayed in a single LC-MS run, and a total of 4 runs in this second phase. However, the EDF stays constant with different variance components’ ratios. This is because the runs and animals are orthogonal to each other; hence the information of either random effects stays intact.

The second example consists of a completely randomised design with 8 animals and 2 treatments for first phase experiment, with these samples arranged in groups of 4 samples assayed across four runs in the second phase experiment. The design of the experiment is as shown in the following table:

Run Ani Tag Trt

1 1 A 114 Con

2 1 B 115 Con

3 1 C 116 Dis

4 1 D 117 Dis

5 2 C 114 Dis

6 2 D 115 Dis

7 2 A 116 Con

8 2 B 117 Con

9 3 E 114 Dis

10 3 F 115 Dis

11 3 G 116 Con

12 3 H 117 Con

13 4 G 114 Con

14 4 H 115 Con

15 4 E 116 Dis

16 4 F 117 Dis

The allocation of the disease statuses to the run and tag is shown in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Run | Tag | | | |
| 114 | 115 | 116 | 117 |
| 1 | Con | Con | Dis | Dis |
| 2 | Dis | Dis | Con | Con |
| 3 | Dis | Dis | Con | Con |
| 4 | Con | Con | Dis | Dis |

The allocation of the animals to the run and tag is written in table below

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Run | Tag | | | |
| 114 | 115 | 116 | 117 |
| 1 | A | B | C | D |
| 2 | C | D | A | B |
| 3 | E | F | G | H |
| 4 | G | H | E | F |

The following code is for generating the simulated data,

run.eff = rnorm(nlevels(design$Run), mean = 0, sd = sqrt(gamma.run \* 1))

ani.eff = rnorm(nlevels(design$Ani), mean = 0, sd = sqrt(gamma.ani \* 1))

trt.eff = runif(nlevels(design$Trt), 0, 2)

tag.eff = runif(nlevels(design$Tag), 0, 1)

res.eff = rnorm(nrow(design), mean = 0, sd = 1)

gm = 10

y = gm + with(design, run.eff[Run] + ani.eff[Ani] + tag.eff[Tag] + trt.eff[Trt]) + res.eff

The gamma.run and gamma.ani are the variance components’ ratios of the runs and animals to the measurement error, respectively. Some disease status and tag effects have been included which allows the simulated dataset to be more realistic.

First function that was applyied to the simulated datasets and design is summary.aov.twoPhase(). The inputs of this function are the simulated dataset mentioned above and the block and treatment structures. The output from the summary.aov.twoPhase() function contains the mean squares in the last column of the random effects table with the variance component structure of the expected mean squares. An example of the output from one dataset can be written as

DF e Ani Run MS

Between Run

Between Ani 1 1 2 4 2.83229

Residual 2 1 0 4 1.41482

Within

Between Ani

Trt 1 1 2 0 5.67374

Tag 1 1 2 0 0.011

Residual 4 1 2 0 1.48418

Residual

Tag 2 1 0 0 0.70796

Residual 4 1 0 0 0.56889

The second function is getVcEDF(). This function takes the ANOVA table generated by the previous function and then generates two tables: one table consists of effective degrees of freedom and the other table contains the variance component estimates. Three types of variance component estimates are produced; these are REML.var.comp, which are the REML-based variance component estimates, LC.var.comp, which are the variance component estimates computed from the linear combination of the expected mean squares, REAL.var.comp, which are the expected variance components. The REML.EDF, LC.EDF and REAL.EDF are the effective degrees of freedom calculated based on the Satterthwaite’s approximation using the three different variance component estimates mentioned.

An example of the output table from getVcEDF() function from one dataset can be shown as

$Stratum

DF e Ani Run MS REML.EDF LC.EDF REAL.EDF

Between RunBetweenAni 1 1 2 4 2.83229 4.389051 4.325900 2.000000

Between RunResidual 2 1 0 4 1.41482 2.333087 2.330261 2.666667

WithinBetweenAniResidual 4 1 2 0 1.48418 4.323423 4.338309 4.571429

WithinResidualResidual 4 1 0 0 0.56889 4.041491 4.046360 4.571429

$Var.comp

LC.var.comp REML.var.comp REAL.var.comp

e 0.5688900 0.5637345 1

Ani 0.4576450 0.4790060 0

Run 0.2114825 0.2306950 0

The following figure plots the residual EDF in the between animals stratum based on the 1000 simulated datasets from the plotEDF() function. In the following plot, each panel shows the EDF for a fixed ratio of the between runs variance component relative to the between measurement variance component. The x-axis, within a panel, corresponds to the VC ratios of between animals and ME and the y-axis denotes the EDF. The three methods of calculating the VCs, namely REML, linear combination (LC) and expected VC, are colour coded black, red and blue, respectively. In this set of plots, no adjustment is made to estimated VCs which are negative.



**Comparing the three methods:**

The EDF generated from the VC estimates of REML and linear combinations are only similar when the VC ratios of run and ME are larger than 1

When the VC ratios of run and ME are smaller than 1, REML produced the highest EDF, the LC produced the second highest EDF and expected VC produced the lowest EDF.

When the VC ratios of run and ME are greater than or equal to 1, the REML and LC methods produced higher EDF compared to the expected VC.

**EDF convergence:**

The EDF converged at 5 as the VC ratio between animals and ME increases to infinity. For the plots with the VC ratios of run and ME are larger than or equal to 10, the convergence was not shown because the VC ratios between animals and ME used here are not large enough.

When the ratio of the run VC relative to the ME VC is larger than or equal to 10, the EDF converges at 4 as the as the VC ratio between animals and ME tend to almost zero.

When the VC ratio of run and ME equals 1, the EDF converges at 4.2 for the REML and LC methods and 4 as the expected VC ratio between animals and ME tend to almost zero.

When the VC ratio of run and ME equals to 0.1, the EDF converges at 4.3 as the expected VC ratio between animals and ME tend to almost zero.

When the VC ratios of run and ME are smaller than or equal to 0.01, the EDF converges at 4.6 as the expected VC ratio between animals and ME tend to almost zero.

When the VC ratios of run and ME are smaller than or equal to 0.1, the convergences of EDF from REML and LC methods are not very obvious as the VC ratio between animals and ME tend to almost zero. This is likely due to the negative VC that is estimated when VC ratios are extremely low using the REML and LC methods.

The following figure plots the EDF of the residual in the between animals stratum based on the 1000 datasets from the plotEDF() function. The following plots consist of nine panels, where each panel contain the EDF plot from variance components (VC) ratio of between the runs and measurement errors (ME). The x-axis shows the VC ratios of between animals and ME and the y-axis denotes the EDF. Three methods, i.e. REML, linear combination (LC) and expected VC, are colour coded by black, red and blue, respectively. In this set of plots, if the estimated VC are negative, the negative VC are adjusted to zero.



**Comparing three methods:**

The EDF from REML and LC are very similar. The only exception is that the EDF of LC method is slightly larger than the REML method when the VC ratio of run and ME equals to 1 and 10.

When the VC ratios of run and ME are larger than or equal to one, the REML and LC methods produced the higher EDF compares to the EDF of the expected VC.

When the VC ratios of run and ME are smaller than or equal to 0.01, the REML and LC methods produce the lower EDF compares to the EDF of the expected VC.

When the VC ratio of run and ME equals to 0.1, the REML and LC methods produce the higher EDF compares to the EDF of the expected VC when the VC ratios of animal and ME are smaller than or equal to 0.1. The REML and LC methods produce the lower EDF compares to the EDF of the expected VC when the VC ratios of animal and ME are larger than 0.1.

**EDF convergence:**

The EDF converged at 5 as the VC ratio between animals and ME increases to infinity. For the plots with the VC ratios of run and ME are larger than or equal to 10, the convergence was not shown because the VC ratios between animals and ME used here are not large enough.

When the variance component ratios of run and measure error are larger or equals to 10, the EDF converged at 4 as the as the variance component ratio between animals and measurement errors tend to almost zero.

When the VC ratio of run and ME equals to 1, the EDF are converged at 4.2 for the REML and LC methods and 4 for the expected VC as the VC ratio between animals and ME tend to almost zero.

When the VC ratio of run and ME equals to 0.1, the EDF are converged at 4.4 for the REML and LC methods and 4.3 for the expected VC as the VC ratio between animals and ME tend to almost zero.

When the VC ratios of run and ME are smaller than or equals to 0.01, the EDF are converged at 4.5 for the REML and the LC methods and 4.6 for the expected VC as the VC ratio between animals and ME tend to almost zero.

The following figure plots the EDF of the residual in the between animals stratum based on the 1000 datasets from the plotEDF() function. In this set of plots, the method called “pool-the-minimum-violator” is used to adjust the negative variance component (VC) estimates (Thompson, 1963). Notice that these EDF plots are very similar to the previous plots, where the negative VC are adjusted to zero, despite different the simulated datasets are used.

****

**Appendix**

We will show here, mathematically, that the expectation of the second derivative of the likelihood function is equal to the DF divided by the twice the square of the MS. Suppose there are *m* MSs in the ANOVA table, and these MS are assumed to have a chi-square distribution. Let these MS be denoted by , the distribution can be written as,

where the denotes the expected MS and is the DF for MS . The likelihood function can then be shownto be

L = constant - .

The first derivative with respect to , also known as the score function, can then be written as

and the expectation of the negative of the second derivative written as

As, the expected Fisher’s information matrix for the MS is the diagonal matrix containing , hence, the MS and DF can be extracted from the ANOVA table to generate Fisher’s information matrix.