

Diamond plots: a tutorial to introduce a visualisation tool that facilitates interpretation and comparison of multiple sample estimates while respecting their inaccuracy

Gjalt-Jorn Y. Peters^{1, 2*}

¹ Department of Statistics & Methodology
Faculty of Psychology & Education Science
Open University

² Department of Work & Social Psychology
Faculty of Psychology & Neuroscience
Maastricht University

* corresponding author: Gjalt-Jorn Peters

Address: PO Box 2960
6401 DL Heerlen, the Netherlands

E-mail: gjalt-jorn@behaviorchange.eu

Diamond plots: a tutorial to introduce a visualisation tool that facilitates interpretation and comparison of multiple sample estimates while respecting their inaccuracy

Abstract

Although a shift from a focus on null hypothesis significance testing to reporting effect sizes and confidence intervals has been advocated for decades, researchers have been slow to implement this shift. One of the reasons may be that working with confidence intervals is interpreted as inconvenient. Diamond plots are a visualisation technique to ameliorate this disadvantage. The current paper introduces an implementation of diamond plots in the free and open source software R. This implementation is flexible and designed to be accessible to researchers that are not used to working with R as well. The current paper also includes a tutorial to enable researchers to start producing diamond plots themselves with minimal effort. Combining a shift from reporting point estimates and confidence intervals in tables to using diamond plots with full disclosure enables presenting reports in a readable manner without loss of detail.

In 2014, the new editor of the journal *Basic and Applied Social Psychology* (BASP) announced a number of laudable policies (Trafimow, 2014), and in 2015, those policies were implemented as BASP journal policy (Trafimow & Marks, 2015, 2016). The journal no longer requires the use of frequentist inferential statistics, banning null hypothesis significance testing (NHST) in favour of descriptive statistics and the requirement of large sample sizes. This shift is consistent with the considerable criticism that has been directed at NHST over the years (Cohen, 1990; Cumming, 2014; Cumming & Finch, 2001; Denis, 2003; Gardner & Altman, 1986; Gigerenzer, Krauss, & Vitouch, 2004), or rather, at the way NHST is applied by most researchers (Morey, Rouder, Verhagen, & Wagenmakers, 2014; Wagenmakers, Wetzels, Borsboom, & van der Maas, 2011). Instead, reporting effect size estimates and confidence intervals is often suggested (as is adopting a Bayesian approach, but many researchers are familiar with confidence intervals, and few with Bayesian approaches, see e.g. Gelman, 2008). Supplementing (effect size) estimates with confidence interval bounds is not only much more informative, it also facilitates inferences.

This desirability of confidence intervals as complementing (or sometimes replacing) NHST and point estimates will henceforth be taken as a given. Readers interested in the argument can consult the references in the previous paragraph. The goal of the current contribution is to facilitate taking this perspective seriously. Specifically, a shift to confidence intervals is not without risk and carries new challenges which may prohibit researchers who decide to consistently utilize confidence intervals in their analyses. In the current paper, I briefly discuss these and present a convenient visualisation tool that can assist in ameliorating these problems.

The example study

I will draw examples from a study into the determinants of using a high dose of ecstasy (for more details, see Crutzen, Peters, & Noijen, 2017). In this study, a number of beliefs regarding using a high dose of ecstasy were measured using a seven-point scale. Five of these beliefs were selected for this example: the belief that using a high dose of ecstasy causes less versus more hallucinations; makes time pass slower versus faster; degrades versus improves the memory of the experience; is worse versus better for one's health; and yields a worse versus better experience. In addition, the direct measures of attitude and intention were used (for details, see Crutzen et al., 2017). These examples are based on 201 participants. The analysis script and output files are available at the Open Science Framework at https://osf.io/9w8yv/?view_only=924e7a3901194a06a204754a735f5aa2.

Challenges when working with confidence intervals

The first problem is perhaps more of a challenge than a problem. Working with confidence intervals means that a researcher has to simultaneously interpret three times as many statistics when surveying analysis results. For most researchers, comparing a list of point estimates in a table is quite manageable. However, adding lower and upper bounds makes this endeavour considerably more challenging.

For example, for the five beliefs related to using a high dose of ecstasy, the respective means are 5.44, 5.58, 2.85, 2.07, and 3.76, and their correlations with attitude are -.18, -.02, .35, .48, .66. This list of five values is quite straightforward to evaluate. However, when complemented with the lower and upper bounds of the confidence intervals, comparing these estimates requires evaluating fifteen values simultaneously, which carries a considerably higher load (see Table 1). Some may even be tempted to simply ignore the columns with lower and upper bounds and only evaluate the associations on the basis of

their point estimates. It would be useful if a method were available to facilitate interpretation of multiple confidence intervals simultaneously.

Table 1: Correlation coefficients and their 95% confidence intervals for the associations between the beliefs and the direct measure of attitude, all regarding using a high dose of ecstasy.

	lower	r	upper
A high dose causes more hallucinations	-.31	-.18	-.05
A high dose makes time pass faster	-.16	-.02	.12
A high dose improves my memory	.23	.35	.47
A high dose is better for my health	.37	.48	.58
A high dose yields a better experience	.58	.66	.73

The second problem is based on the unfortunate reality that despite their advantages, confidence intervals are still subject to sample-to-sample variability. This means that the point estimate that the confidence interval is computed from, and therefore also the lower and upper bounds, vary from sample to sample. The degree to which the sample point estimates can be expected to differ from the population value and from sample to sample is a function of the width of their respective sampling distributions. If that sampling distribution is wide, a statistic's value can vary considerably from sample to sample, and therefore, so can the confidence interval bounds. Common sample sizes range in the dozens or sometimes low hundreds, and such sample sizes often yield very wide effect size sampling distributions (e.g. for correlation coefficients, Moinester & Gottfried, 2014; or Cohen's d , Peters & Crutzen, 2017). Given these widths, accuracy is often an illusion, and treating estimates as if they were accurate can be misleading. There is often no reason to

place particular trust in the values obtained from any specific sample. The lower and upper bounds of confidence intervals, though quantitative, should not be perceived as enabling conclusions in and of themselves (Gelman & Hennig, 2015; Greenland et al., 2016).

However, because the lower and upper bounds are expressed as numbers, researchers may be tempted to apply criteria to these values. For example, researchers could consider confidence intervals that do not contain the value 0 as indicative of 'a true population effect'; and conversely, interpret intervals that do contain the value 0 as indicative of 'no true population effect'. This, of course, would amount to applying NHST, and the same criticisms apply (with respect to this latter point, also see Lakens, 2017). Therefore, when working with confidence intervals, it would be useful if a method was available that is less susceptible to this temptation.

Visualisation

It has been argued that results of quantitative research are better presented visually than quantitatively (Gelman, Pasarica, & Dodhia, 2002), and even that harder sciences rely more on visualisation where softer sciences rely more on tabulation (Smith, Best, Stubbs, Archibald, & Roberson-Nay, 2002). Proper visualisation has the potential to solve both problems. Means are often plotted using bar plots or dynamite plots, but these are perceptually misinterpreted (Newman & Scholl, 2012). Another common visualisation plots means using a line, point or box, adding error bars to plot the confidence intervals, often supplemented with 'whiskers'. This is, for example, what dot whisker plots do. Dot whisker plots have already been made easily accessible for plotting regression coefficients (Solt & Hu, 2016). However, a problem of this popular approach is that it draws attention to the point estimate and the confidence interval bounds. While this makes sense given that these

are the values being plotted, it also invites researchers to compare those bounds. As explained before, these bounds vary from sample to sample; their values in any given sample are largely due to chance. There is no reason to place much trust in the exact values of these bounds in any specific sample, and in fact, of all values in the confidence interval, the bounds could be argued to represent the *least* likely population values. It would be better if a visualisation would not draw the audience's attention to the confidence interval bounds, instead subtly drawing attention away.

Such a visualisation technique has already been widely implemented in the forest plots used in meta-analyses and generated, for example, by the Free and Open Source Software (FOSS) R package metafor (Viechtbauer, 2010). In such forest plots, although dot whisker plots are commonly used to plot the effect size estimates and confidence intervals from each included study, the aggregate effect size is often plotted using a diamond shape. Diamonds solve the aforementioned problems: they enable easy comparison and do not draw attention to the confidence interval bounds. Because their bounds are not clearly distinguished they hinder confident statements about which diamonds represent similar or different estimates in situations where such confidence is not warranted. Diamonds afford conclusions that are more commensurate with the accuracy permitted by the data than dots and whiskers. In a way, diamond plots encourage a level of scientific modesty that is fitting given the dataset.

How to obtain diamond plots

Diamond plots have been made available in the FOSS R package userfriendlyscience (Crutzen & Peters, 2015; Peters, 2014, 2017). R (R Development Core Team, 2017) is a powerful and versatile FOSS statistical package that is frequently used together with the FOSS graphical user interface provided by RStudio (RStudio Team, 2016). Once these have

been installed, packages can be downloaded and installed using the ‘install.packages’ command. Once installed to the local computer, a package can be loaded using the ‘require’ command (although installing is only necessary once, packages need to be reloaded every time a new session is started):

```
install.packages('userfriendlyscience');  
require('userfriendlyscience');
```

There are several ways to obtain diamond plots. For researchers unfamiliar with R or in situations where raw data are not available, means, standard deviations, sample sizes and variable labels can be specified directly:

```
meanSDtoDiamondPlot(labels = c('Hallucination',  
                                'Time',  
                                'Memory',  
                                'Health',  
                                'Experience'),  
                    means = c(5.44, 5.58, 2.85, 2.07, 3.76),  
                    sd = c(1.19, 1.42, 1.4, 1.14, 1.63),  
                    ns = c(201, 201, 201, 201, 201));
```

This will yield the diamond plot shown in panel A of Figure 1. Diamond plots can be adjusted by using the R package that implements a grammar of graphics for plotting, ggplot2 (because the functions return a ggplot2 plot; Wickham, 2009). This makes it possible to, for example, adjust the x axis to cover the full width of the answer scales (from 1 to 7), add a title, and change the x axis label (see panel B of Figure 1):

```
meanSDtoDiamondPlot(labels = c('Hallucination',  
                                'Time',  
                                'Memory',  
                                'Health',  
                                'Experience'),  
                    means = c(5.44, 5.58, 2.85, 2.07, 3.76),  
                    sd = c(1.19, 1.42, 1.4, 1.14, 1.63),  
                    ns = c(201, 201, 201, 201, 201)) +  
  scale_x_continuous(limits=c(1, 7), breaks=1:7) +  
  xlab('Means and 95% confidence intervals') +  
  ggtitle('A diamond plot');
```

In most situations, however, raw data will be available, so it will be more convenient to construct the diamond plots from those raw data.

Diamond plots based on raw data

Data can be loaded into R using the 'getDat' command:

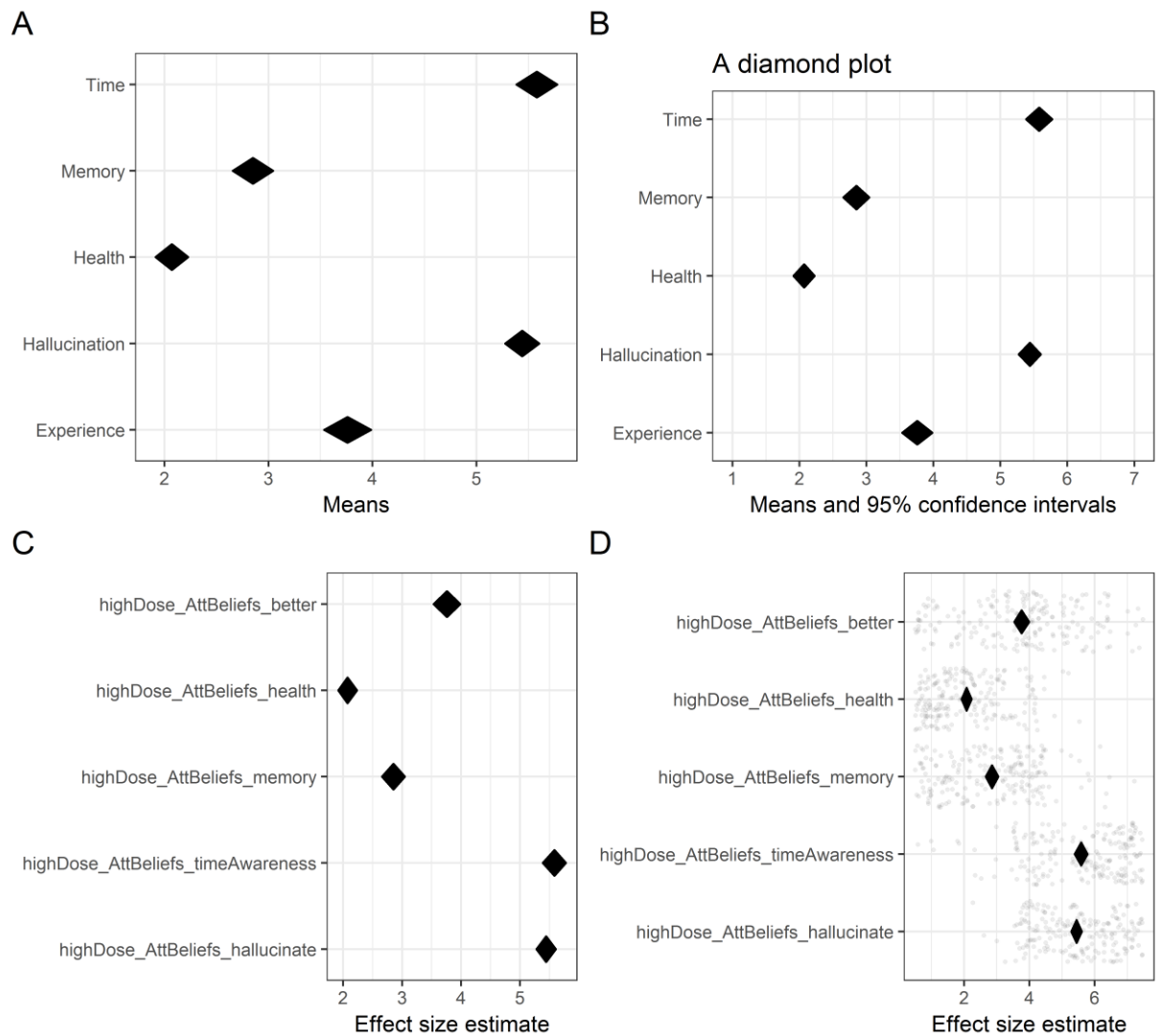
```
getDat();
```

This command will open a dialogue allowing the user to select a file, for example an SPSS data file. This data file is then loaded into R and made available under the name 'dat'.

To reproduce the diamond plot from the first example, the command 'meansDiamondPlot' can be used (see panel C of Figure 1):

```
meansDiamondPlot(dat, items=c('highDose_AttBeliefs_hallucinate',  
                              'highDose_AttBeliefs_timeAwareness',  
                              'highDose_AttBeliefs_memory',  
                              'highDose_AttBeliefs_health',  
                              'highDose_AttBeliefs_better'),  
                 showData=FALSE);
```

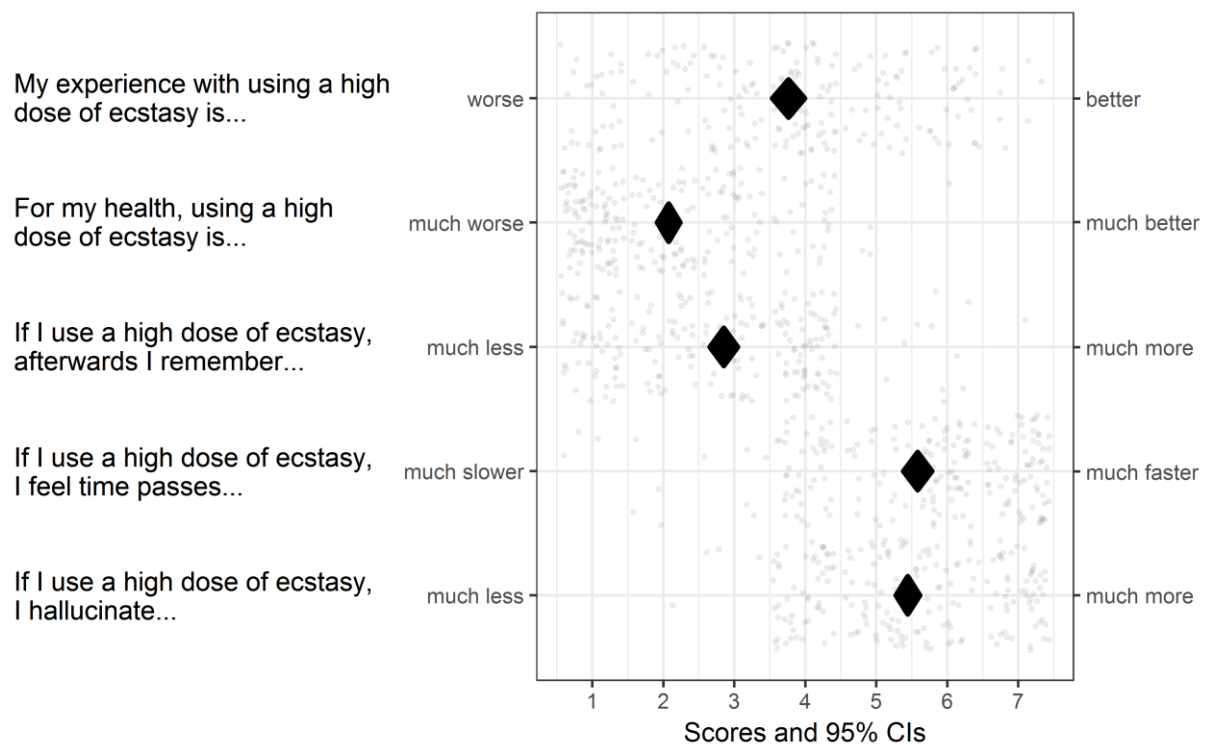
This last argument hides the data points to obtain a reproduction of the plot in panel A. By default, the data points are shown to facilitate interpretation of the summary statistics and confidence intervals. The diamond plot that would be obtained if this last argument were omitted is shown in panel D of Figure 1. All these plots can be further customized in the same way as the plot in panel B. For example, labels can be supplied that will replace the variable names.



[INSERT FIGURE 1 HERE (included for the convenience of peer review; caption text is “Figure 1: Four diamond plots representing univariate distributions.”)]

In some situations, for example when visualising beliefs like in this example data set, each operationalisation uses a different anchor. In this case, ‘biAxisDiamondPlot’ can be used, specifying the name of the data set (or, in R vocabulary, ‘data frame’), the names of the variables to show, the anchors to show at the left y axis, the anchors to show at the right y axis, and the ‘sub questions’ or variable labels. An example is shown in Figure 2, and the R

command to produce this plot is available in the Open Science Framework repository for this paper (see https://osf.io/9w8yv/?view_only=924e7a3901194a06a204754a735f5aa2).



[INSERT FIGURE 2 HERE (included for the convenience of peer review; caption text is

“Figure 2: A diamond plot with two y axes.”]

Diamond plots of associations

Although an easy method for combining data points, point estimates, and confidence intervals into one figure can be convenient in many situations, the improvements over dot whisker plots are relatively minor. However, the same logic underlying diamond plots of univariate distributions can be applied to visualisations of associations. This makes it possible to compare associations efficiently.

For example, the following command produces a diamond plot for the associations between the five beliefs in the example and the direct measure of attitude (see Figure 3, panel A):

```
associationsDiamondPlot(dat,
                        covariates=c('highDose_AttBeliefs_hallucinate',
                                     'highDose_AttBeliefs_timeAwareness',
                                     'highDose_AttBeliefs_memory',
                                     'highDose_AttBeliefs_health',
                                     'highDose_AttBeliefs_better'),
                        criterion='highDose_attitude');
```

Such diamond plots do not only facilitate comparing associations among multiple covariates. It is also possible to stack multiple layers, which facilitates comparing associations with multiple criteria (dependent variables), using the argument 'returnLayerOnly'. The layers can be distinguished by specifying a different color for the diamonds in each layer using the argument 'color', and a layer's diamonds can be made transparent by passing an alpha of less than 1 (where 1 corresponds to full opacity and 0 corresponds to full transparency; see panel B in Figure 3):

```
associationsDiamondPlot(dat, covariates=c('highDose_AttBeliefs_hallucinate',
                                          'highDose_AttBeliefs_timeAwareness',
                                          'highDose_AttBeliefs_memory',
                                          'highDose_AttBeliefs_health',
                                          'highDose_AttBeliefs_better'),
                        criterion='highDose_attitude',
                        alpha=.5, color = 'blue') +
associationsDiamondPlot(dat, covariates=c('highDose_AttBeliefs_hallucinate',
                                          'highDose_AttBeliefs_timeAwareness',
                                          'highDose_AttBeliefs_memory',
                                          'highDose_AttBeliefs_health',
                                          'highDose_AttBeliefs_better'),
                        criterion='highDose_intention',
                        alpha=.5, color='red',
                        returnLayerOnly = TRUE);
```

A number of functions exist to further facilitate interpretation and comparison. For example, the argument 'generateColors' can be used to specify two or more colors that are then used to generate a gradient of diamond colors that correspond to the association strengths. This argument can be combined with argument 'fullColorRange' to anchor the boundary colors. For example (see panel C in Figure 3):

```
associationsDiamondPlot(dat, covariates=c('highDose_AttBeliefs_hallucinate',
```

```

        'highDose_AttBeliefs_timeAwareness',
        'highDose_AttBeliefs_memory',
        'highDose_AttBeliefs_health',
        'highDose_AttBeliefs_better'),
    criterion='highDose_attitude',
    generateColors=c("red", "green"),
    fullColorRange=c(-1, 1));

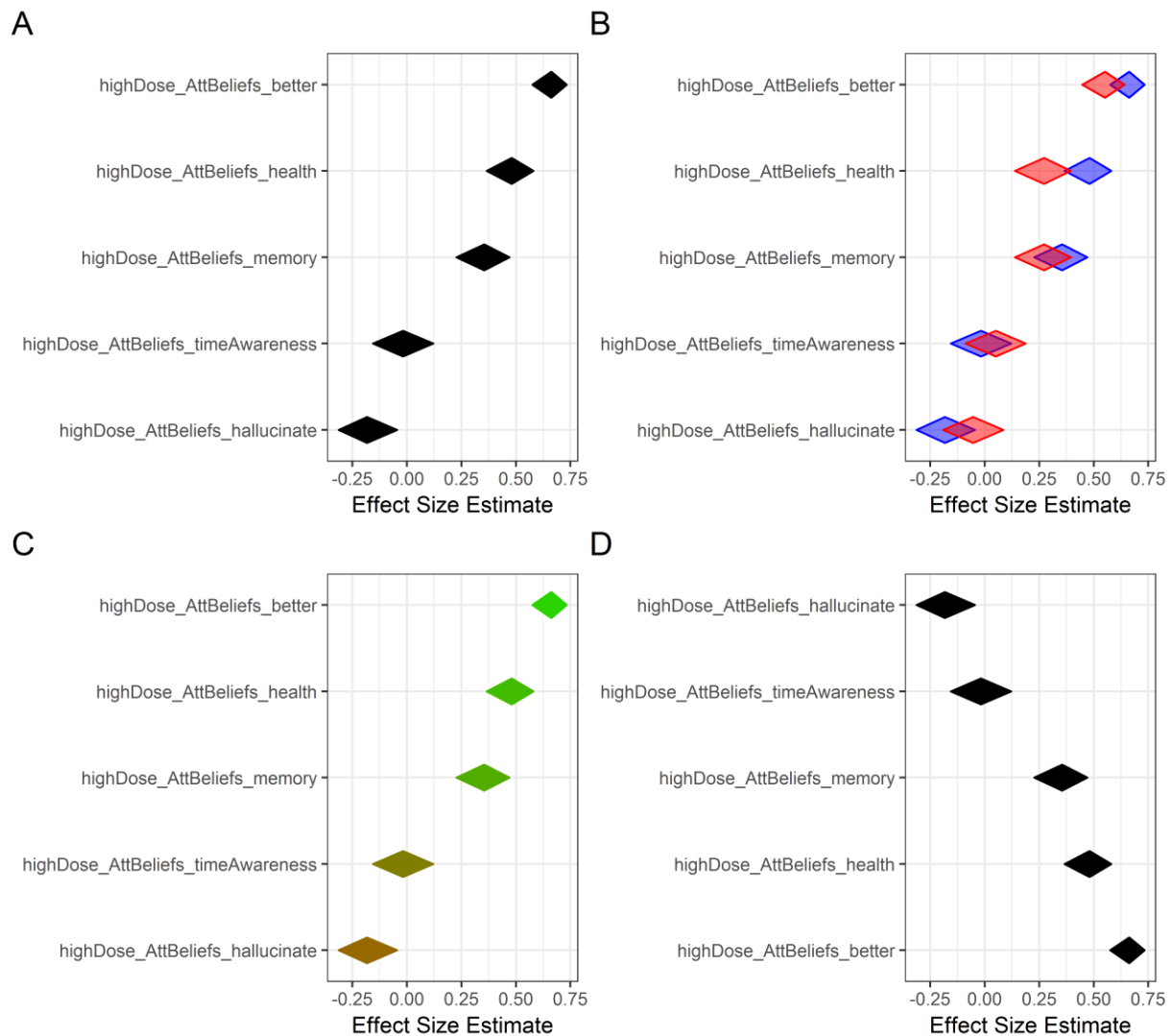
```

It is also possible to use argument 'decreasing' to automatically sort the rows ascendingly (decreasing=TRUE) or descendingly (decreasing=FALSE; see panel D in Figure 3):

```

associationsDiamondPlot(dat,
    covariates=c('highDose_AttBeliefs_hallucinate',
        'highDose_AttBeliefs_timeAwareness',
        'highDose_AttBeliefs_memory',
        'highDose_AttBeliefs_health',
        'highDose_AttBeliefs_better'),
    'highDose_attitude',
    decreasing=FALSE);

```



[INSERT FIGURE 3 HERE (included for the convenience of peer review; caption text is “Figure 3: Diamond plots for associations.”)]

Other applications and extensions

Both types of diamond plots (i.e. diamond plots visualising univariate distributions and diamond plots visualising associations) can also be combined, which can be particularly insightful when conducting determinant studies. This approach to confidence interval-based estimation of relevance of determinants is discussed in detail by Crutzen, Peters & Noijen (2017).

Finally, it is also possible to specify diamond plots manually. This is done by directly calling the `diamondPlot` function, passing a data frame and specifying which columns contain the lower bound, point estimate, and upper bound of the confidence intervals. This makes it possible to generate diamond plots for all sets of confidence intervals. It also enables researchers who are unfamiliar with R to produce diamond plots for confidence intervals obtained from other statistical software. By using the `diamondPlot` function directly, the only three commands necessary are those for installing and loading the `userfriendlyscience` package (see above) and the ‘`diamondPlot`’ function. For example, the diamond plot in panel A of Figure 1 can be reproduced as follows:

```
diamondPlot(data.frame(ci.lo = c(-.31, -.16, .23, .37, .58),
                      es = c(-.18, -.02, .35, .48, .66),
                      ci.hi = c(-.05, .12, .47, .58, .73)),
            yLabels=c('Hallucinate', 'Time', 'Memory', 'Health', 'Experience'));
```

More details on any of these functions can be obtained by using R’s native help system: prepending a question mark to any of the function names shows the help page for that function (e.g., using the command ‘`?diamondPlot`’). In addition, the source code for any of

the functions can be obtained by entering the function name (e.g. 'diamondPlot'), or by visiting the package GitHub repository.

There are several ways of storing plots produced in R to a file. The easiest method is by using a function from the `ggplot2` package called 'ggsave'. First, store the diamond plot by prepending the command with the desired name and the R assignment operator, for example:

```
a <- diamondPlot(data.frame(ci.lo = c(-.31, -.16, .23, .37, .58),
                             es = c(-.18, -.02, .35, .48, .66),
                             ci.hi = c(-.05, .12, .47, .58, .73)),
                 yLabels=c('Hallucinate', 'Time', 'Memory', 'Health', 'Experience'));
```

This object ('a') can then be printed using `ggsave`. For Windows PC's, the following command will store the plot in a file called 'figure.png' on the user's desktop, in a size of 10 by 10 centimetres at 300 dots per inch (dpi):

```
ggsave(plot = a,
        file = path.expand("~/../Desktop/figure.png"),
        width=10, height=10, units='cm', dpi=300);
```

Note that in R, it is usually easiest to use forward slashes instead of backward slashes when specifying paths (i.e. directories and filenames).

Conclusion

Diamond plots provide a versatile, powerful way of visualising estimates computed from samples. They facilitate consistent application of confidence intervals by removing some of the problems associated with interpreting many confidence intervals simultaneously. In addition, diamond plots impede application of 'bright-line' decision criteria, thereby promoting statistical best practice (Wasserstein & Lazar, 2016). Their implementation in the Free and Open Source Software statistical package R makes them widely available, and the possibility of generating diamond plots directly from estimates obtained from other

statistical packages means that even researchers unfamiliar with the R software can easily produce diamond plots.

Using diamond plots to describe univariate distributions and bivariate associations and dot whisker plots (Solt & Hu, 2016) for results of regression analyses even make it possible to omit tables completely. This would further foster a cautious approach to statistical analysis, discouraging confident statements where empirical support is tenuous. Of course, traditional reporting customs would be incompatible with such a shift from reporting quantitative estimates to visualisation. However, practising full disclosure means that instead of less, more data are available for meta-analysis and interested researchers (Crutzen, Peters, & Abraham, 2012; Fuller, Pearson, & Peters, 2013; Peters, Abraham, & Crutzen, 2012; Peters, Kok, Crutzen, & Sanderman, 2017). This would combine the best of both worlds: such study reports enable accurate reproduction of analyses and data synthesis, while promoting appropriate modesty when drawing conclusions in the report itself. Using diamond plots in combination with full disclosure results in clear, readable study reports without any loss of detail.

References

- Cohen, J. (1990). Things I Have Learned (So Far). *American Psychologist*, 45, 1304–1312.
- Crutzen, R., & Peters, G.-J. Y. (2015). Scale quality: alpha is an inadequate estimate and factor-analytic evidence is needed first of all. *Health Psychology Review*.
<http://doi.org/10.1080/17437199.2015.1124240>
- Crutzen, R., Peters, G.-J. Y., & Abraham, C. (2012). What about trialists sharing other study materials? *BMJ*, 345, e8352–e8352. <http://doi.org/10.1136/bmj.e8352>
- Crutzen, R., Peters, G.-J. Y., & Noijen, J. (2017). How to Select Relevant Social-Cognitive Determinants and Use them in the Development of Behaviour Change Interventions? Introducing Confidence Interval-Based Estimation of Relevance. *PsyArXiv*. Retrieved from <http://osf.io/5gnmz>

- Cumming, G. (2014). The New Statistics: Why and How. *Psychological Science*, 25(1), 7–29. <http://doi.org/10.1177/0956797613504966>
- Cumming, G., & Finch, S. (2001). A Primer on the Understanding, Use, and Calculation of Confidence Intervals that are Based on Central and Noncentral Distributions. *Educational and Psychological Measurement*, 61(4), 532–574. <http://doi.org/10.1177/0013164401614002>
- Denis, D. J. (2003). Alternatives to Null Hypothesis Significance Testing. *Theory & Science*, 4(1), 2.
- Fuller, T. E., Pearson, M., & Peters, J. L. (2013). Transparent Reporting, the Foundation for Full Disclosure. *The European Health Psychologist*, 15(3), 67–68. <http://doi.org/10.2105/ajph.94.3.361>
- Gardner, M. J., & Altman, D. G. (1986). Statistics in Medicine Confidence intervals rather than P values: estimation rather than hypothesis testing. *British Medical Journal (Clinical Research Ed.)*, 292(6522), 746–750. <http://doi.org/10.1136/bmj.292.6522.746>
- Gelman, A. (2008). Objections to Bayesian statistics. *Bayesian Analysis*, 3(3), 445–450. <http://doi.org/10.1214/08-BA318>
- Gelman, A., & Hennig, C. (2015). Beyond subjective and objective in statistics * Our proposal, (June).
- Gelman, A., Pasarica, C., & Dodhia, R. (2002). Let's Practice What We Preach. *The American Statistician*, 56(2), 121–130. <http://doi.org/10.1198/000313002317572790>
- Gigerenzer, G., Krauss, S., & Vitouch, O. (2004). The Null Ritual What You Always Wanted to Know About Significance Testing but Were Afraid to Ask. In D. Kaplan (Ed.), *The Sage handbook of quantitative methodology for the social sciences* (pp. 391–408). Thousand Oaks, CA: Sage.
- Greenland, S., Senn, S. J., Rothman, K. J., Carlin, J. B., Poole, C., Goodman, S. N., & Altman, D. G. (2016). Statistical tests, P values, confidence intervals, and power: a guide to misinterpretations. *European Journal of Epidemiology*, 31(4), 337–350. <http://doi.org/10.1007/s10654-016-0149-3>
- Lakens, D. (2017). Equivalence Tests: A Practical Primer for t-Tests, Correlations, and Meta-Analyses. <http://doi.org/doi.org/10.1177/1948550617697177>
- Moinester, M., & Gottfried, R. (2014). Sample size estimation for correlations with pre-specified confidence interval. *The Quantitative Methods of Psychology*, 10(2), 124–130. Retrieved from <http://tqmp.org/RegularArticles/vol10-2/p124/>
- Morey, R. D., Rouder, J. N., Verhagen, J., & Wagenmakers, E.-J. (2014). Why Hypothesis Tests Are Essential for Psychological Science: A Comment on Cumming. *Psychological Science*, 25(6), 1289–90. <http://doi.org/10.1177/0956797614525969>
- Newman, G. E., & Scholl, B. J. (2012). Bar graphs depicting averages are perceptually misinterpreted: the within-the-bar bias. *Psychonomic Bulletin & Review*, 19(4), 601–7. <http://doi.org/10.3758/s13423-012-0247-5>
- Peters, G.-J. Y. (2014). The alpha and the omega of scale reliability and validity: why and how to abandon Cronbach's alpha and the route towards more comprehensive assessment of scale quality. *The European Health Psychologist*, 16, 56–69.
- Peters, G.-J. Y. (2017). userfriendliscience: Quantitative analysis made accessible. Retrieved from <http://cran.r-project.org/package=userfriendliscience>
- Peters, G.-J. Y., Abraham, C., & Crutzen, R. (2012). Full disclosure: doing behavioural science necessitates sharing. *The European Health Psychologist*, 14, 77–84.

- Peters, G.-J. Y., & Crutzen, R. (2017). Knowing exactly how effective an intervention, treatment, or manipulation is and ensuring that a study replicates: accuracy in parameter estimation as a partial solution to the replication crisis. Retrieved from <http://osf.io/preprints/psyarxiv/cjsk2>
- Peters, G.-J. Y., Kok, G., Crutzen, R., & Sanderman, R. (2017). Health Psychology Bulletin: improving publication practices to accelerate scientific progress. *Health Psychology Bulletin*, 1(1), 1–6. <http://doi.org/10.5334/hpb.2>
- R Development Core Team. (2017). R: A Language and Environment for Statistical Computing. Vienna, Austria. Retrieved from <http://www.r-project.org/>
- RStudio Team. (2016). RStudio: Integrated Development Environment for R. Boston, MA. Retrieved from <http://www.rstudio.com/>
- Smith, L. D., Best, L. a., Stubbs, D. A., Archibald, A. B., & Roberson-Nay, R. (2002). Constructing knowledge: The role of graphs and tables in hard and soft psychology. *American Psychologist*, 57(10), 749–761. <http://doi.org/10.1037//0003-066X.57.10.749>
- Solt, F., & Hu, Y. (2016). dotwhisker: Dot-and-Whisker Plots of Regression Results. Retrieved from <https://cran.r-project.org/web/packages/dotwhisker/vignettes/dotwhisker-vignette.html>
- Trafimow, D. (2014). Editorial. *Basic and Applied Social Psychology*, 36(1), 1–2. <http://doi.org/10.1080/01973533.2014.865505>
- Trafimow, D., & Marks, M. (2015). Editorial. *Basic and Applied Social Psychology*, 37(February), 1–2. <http://doi.org/10.1080/01973533.2015.1012991>
- Trafimow, D., & Marks, M. (2016). Editorial. *Basic and Applied Social Psychology*, 38(1), 1–2. <http://doi.org/10.1080/01973533.2016.1141030>
- Viechtbauer, W. (2010). Conducting Meta-Analyses in R with the metafor package. *Journal Of Statistical Software*, 36(3), 1–48.
- Wagenmakers, E.-J., Wetzels, R., Borsboom, D., & van der Maas, H. L. J. (2011). Why psychologists must change the way they analyze their data: the case of psi: comment on Bem (2011). *Journal of Personality and Social Psychology*, 100(3), 426–32. <http://doi.org/10.1037/a0022790>
- Wasserstein, R. L., & Lazar, N. A. (2016). The ASA's Statement on p-Values: Context, Process, and Purpose. *The American Statistician*, 70(2), 129–133. <http://doi.org/10.1080/00031305.2016.1154108>
- Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. Retrieved from <http://ggplot2.org>