Response by author to interactive comment by RC1 on “Fast sensitivity analysis methods for computationally expensive models with multidimensional output” by Edmund Ryan et al.

Topical Editor Decision: Publish subject to minor revisions (review by editor) (25 May 2018) by Andrea Stenke
Comments to the Author:
Dear Edmund Ryan,

thanks a lot for revising your manuscript according to the referees' comments.

There is one last and probably minor point which needs clarification. Referee #1 brought this to my attention after submission of your author's comment. The referee thinks that you might have misunderstood one of the comments: "In Figures 3 and 4 the scaling appears to be wrong. They say they are plotting sensitivity indices which should be in [0,1], but the color bar indicates that they are in [0,70]. My guess is that the plot is correct and it's a mistake with scaling when they constructed the color bar."

Could you please briefly comment on this or, if necessary, revise the figures accordingly? After this has been clarified I will be happy to accept your paper for publication.

Best regards,
Andrea Stenke

Author’s response:

Thanks for spotting this. In figures 3 and 4, the SIs are given as percentages (i.e. 0-100) but in the manuscript they are given as being in the 0-1 range. To fix this, I have added the following (highlighted in yellow) to the text following equation 1 on page 9 of the manuscript. I add $\times 100$ to equation 2 as well.

$$S_{i,j} = \frac{Var\left[E\left(Y_j|X_i\right)\right]}{Var(Y_j)} \times 100 \tag{1}$$

where X_i is the ith column of the $n \times p$ matrix (i.e. a matrix with n rows and p columns) which stores the n samples of p-dimensional inputs and Y_j is the jth column of the $n \times m$ matrix which stores the corresponding n sets of m-dimensional outputs (table 1). We multiply by 100 so that the SI is given as a percentage. The notation given by $Var(\cdot)$ and $E(\cdot)$ denote the mathematical operations that compute the variance and expectation. The simplest way of computing $S_{i,j}$ is by brute force, but this is also the most computationally intensive (Saltelli et al., 2008).