Comments on ScottKnottESD in response to "An Empirical Comparison of Model Validation Techniques for Defect Prediction Models"

Steffen Herbold

Abstract—In this article, we discuss the ScottKnottESD test, which was proposed in a recent paper "An Empirical Comparison of Model Validation Techniques for Defect Prediction Models" that was published in this journal. We discuss the implications and the empirical impact of the proposed normality correction of ScottKnottESD and come to the conclusion that this correction does not necessarily lead to the fulfillment of the assumptions of the original Scott-Knott test and may cause problems with the statistical analysis.

1 Introduction

In the article "An Empirical Comparison of Model Validation Techniques for Defect Prediction Models" by Tantithamthavorn *et al.* [1], the authors propose Scott-Knott Effect Size Difference (ScottKnottESD) as an extension of the Scott-Knott test [2]. Within this response to the article, we want to comment on this extension and the implications of using ScottKnottESD, as well as on problems that may occur, and give recommendations for using ScottKnottESD.

To this aim, we first summarize the original Scott-Knott test in Section 2 and ScottKnottESD in Section 3. Then, we discuss the implications of the proposed normality correction in Section 4. In Section 5, we proceed to show a small experiment that exemplifies the discussed implications and highlights the impact both on real, as well as on artificial data. Afterwards, we give recommendations on the future use of ScottKnottESD in Section 6. In Section 7, we summarize the feedback from Tantithamthavorn *et al.* which we got when we contacted them. Finally, we conclude our article in Section 8.

2 Summary of the Scott-Knott test

The Scott-Knott test [2] is a statistical procedure for the clustering of siginificantly different results as outcome of an Analysis of Variance (ANOVA) test [3]. ANOVA determines if there are statistically significant differences between groups of populations. As a corollary, the three requirements of ANOVA must be fulfilled if one wants to use the Scott-Knott test.

1) Normality: the residuals of the dependent variables must be normally distributed.
2) Homoscedasticity: the variance of all dependent variables must be the same.
3) Independence of observations: all observations must be independent of each other.

If ANOVA finds a significant difference between the populations, Scott-Knott uses a cluster analysis method to first determine two groups of populations, such that the within group sum of squares of both groups is minimized. This procedure is recursively repeated until the groups are homogeneous, i.e., ANOVA does not detect significantly different populations within a resulting group. For the purpose of our discussion of Scott-Knott in this article, three aspects are relevant: the normality assumption of ANOVA, the homoscedasticity assumption of ANOVA, and the cluster analysis based on minimizing the within group sum of squares.

3 Modifications by ScottKnottESD

ScottKnottESD is identical to the normal Scott-Knott test, except for two changes.

1) Normality correction: prior to the application of the Scott-Knott test, Tantithamthavorn *et al.* propose a log-transformation of the data, such that $x' = log(x + 1)$.
2) Effect size correction: after the application of the Scott-Knott test, Cohen's d [4] is used to merge clusters where the effect size is negligible, i.e., $d < 0.2$.

The rationale behind the first modification is to treat a potential skewness in the variable distribution with the purpose to fulfill the normality assumption. The rationale behind the second modification is that groups of populations with a negligible difference in effect sizes should not be in different clusters, even if the difference is statistically significant. The authors published their implementation as the package ScottKnottESD v1.1 on CRAN [5].

• *S. Herbold is with the University of Goettingen, Institute of Computer Science, Goettingen, Germany.*
E-mail: herbold@cs.uni-goettingen.de

4 IMPLICATIONS OF SCOTTKNOTTESD

From our point of view, the idea to use Cohen's d is very good and can help to improve the interpretation of results by merging clusters and, thereby, achieving a clearer grouping. Therefore, we will not go into greater detail regarding this adoption, as we fully agree with Tantithamthavorn *et al.* that this change is valuable.

The normality correction, on the other hand has several implications of, the impact of which cannot be predicted in general. Therefore, we want to take a closer look at the rationale for the normality correction and the implications on the various steps of ScottKnottESD.

4.1 Rationale behind the normality correction

The authors state that the log-transformation treatment is "a commonly-used transformation technique in software engineering research" based on [6], [7]. We agree that the list of articles, where such a log-transformation is used, is actually quite long. Moreover, the use of log-transformations to deal with skewed features is not limited to defect prediction, but also used for other tasks, e.g., for the analysis of globally distributed software development teams [8] and for expertise modeling [9]. Additionally, log-transformations are also used for other software engineering models to deal with multiplicative relationships, e.g., when calibrating COCOMO II models [10]. However, to the best of our knowledge, no article on defect prediction applied the log-transformation to performance measures. Instead, the log-transformation is applied to the features before using them for machine learning. As a case in point, Tantithamthavorn *et al.* themselves use the log-transformation for features that way in the same article where ScottKnottESD is proposed [1].

From our point of view, the application of log-transformation to performance metrics is fundamentally different from the application to features. For features, the goal is to reduce the skewness, as many machine learning algorithms favor data with less skewness. In the end, one is not interested in the feature, but how the feature helps to predict the outcome. Thus, even if the skewness is not bad or fully treated using the transformation, this only results in a different prediction model that will be evaluated. The evaluation itself is not affected.

However, when performance metrics are transformed, this directly impacts the evaluation of results. Basically, the statistical tests do not evaluate, e.g., AUC anymore, but $log(\text{AUC} + 1)$. This has several implications on a procedure like Scott-Knott.

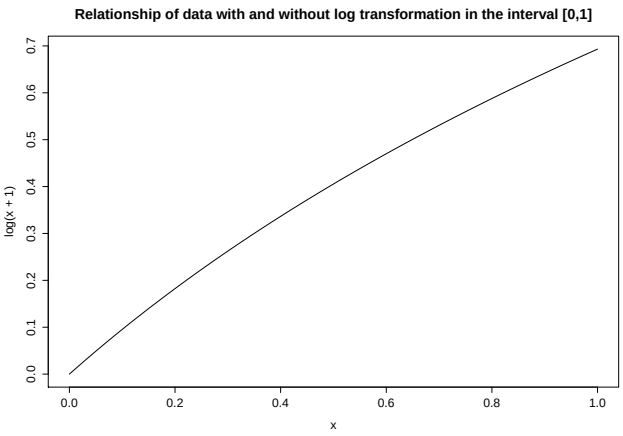

Fig. 1. Change in values due to log-transformation.

4.2 Implications of the normality correction on ANOVA

The first are the implications for ANOVA. The authors state in the paper that ScottKnottESD "makes no assumptions about the underlying distribution" [1]. They argue that this is achieved by the log-transformation, which they refer to as normality correction in this context. However, a log-transformation does not guarantee normality. If data is already normally distributed or if the skewness is not exponential of a normal distribution, the log-transformation will not help with the fulfillment of the normality assumption. In case the data is already normally distributed, the log-transformation may actually change the data distribution such that it is no longer normal.

Moreover, the log-transformation also impacts the variance of the data, and therefore, the homoscedasticity assumption of ANOVA. Figure 1 shows how the values of a performance measure change due to the log-transformation, assuming that the possible values of the performance measure are in the interval $[0, 1]$. The relative distances between the values decrease with growing performance values. Thus, the variance between these high values of a measure will change differently under the transformation than the variance between lower values of a performance measure. How this effects the homoscedasticity is not explored by Tantithamthavorn *et al.*

4.3 Implications of the normality correction on the cluster analysis

The cluster analysis is similarly affected by the log-transformation. Because the relative distances between performance metrics change, the within group sum of squares is also different than without the transformation. Basically, it means that the clusters are not created over the performance metrics, but instead over the log-transformation of the performance metrics. Consequently, the groups may be different with

or without the log-transformation. Hence, it is unclear if the groupings hold for the actual values of the performance metrics, or only for the log-transformations.

4.4 Implications of the normality correction on the effect size correction

The implications on the effect size correction are basically the same as on the cluster analysis. Cohen's d uses both the difference between the mean values, as well as the difference between the standard deviations. Both are changed due to the effect of the log-transformation on the relative distances between the results. Specifically, the differences decrease after the log-transformation reducing the effect size, especially for larger performance values. Thus, Cohen's d may yield a negligible effect size for the log-transformed values, but a larger and not negligible effect size for the actual values of the performance metrics. In this case, clusters which are actually statistically significantly different with a non-negligible effect size my still be merged.

5 Experiments on the impact of the normality correction

We performed some experiments to evaluate whether the implications discussed above are of practical relevance. The complete experiment results are available online and are fully reproducible [11].

5.1 Impact on real defect prediction results

To see if and how the normality correction impacts defect prediction results, we simply took existing defect prediction data we had available from previous experiments. In particular, we took the performance values of 62 cross-project predictions we made with data collected by Jureczko and Madeyski [12]. The 62 products are all not proprietary products from the data set. The predictions are strict cross-project predictions without any transfer learning technique applied. The data was generated as part of a benchmark on cross-project defect prediction [13] and represents the results for the baseline configuration ALL. As classifiers we used Naive Bayes (NB) and C4.5 decision trees (DT). As performance metrics we considered AUC and F-measure. For all statistical tests, we use the significance level of $\alpha = 0.95$, i.e., results are significant if the $p - value < 0.05$.

Please note, that the focus of this article is solely on the impact of the normality correction on the evaluation of prediction results. The actual values of the performance metrics, i.e., whether the results are good or bad, are irrelevant for this article and, therefore, not discussed. Similarly, we do not discuss if NB and DT are good choices for algorithms, and if the way the training data is selected is a good strategy.

	AUC		F-measure	
	NB	DT	NB	DT
No log-transformation	0.0098	0.9609	0.0299	0.7824
With log-transformation	0.0009	0.7215	0.1263	0.5531

TABLE 1
$p - values$ of the Shapiro Wilk test for normality.

	AUC		F-measure	
	NB	DT	NB	DT
No log-transformation	0.0129	0.0097	0.0242	0.0181
$p - value$	0.5993		0.4747	
With log-transformation	0.0047	0.0039	0.0144	0.0098
$p - value$	0.9022		0.2364	

TABLE 2
Variances of the data and $p - values$ of Levene's test for variance homogeneity.

5.1.1 Impact on the normality assumption

To test for normality of the data, we used the Shapiro-Wilk test [14]. The null hypotheses of the test is that the data is normally distributed. The null hypothesis is rejected if $p-value < 0.05$. Table 1 shows the results of the test. The results with the DT are both normally distributed, for NB both results are not normally distributed. The log-transformation does not solve the problems with the normality assumption. While the result of NB with F-measure is now normally distributed, the results with NB and AUC are still not normally distributed. The $p - value$ was actually reduced, meaning the results are even more significantly non-normal. Similarly, the $p-value$ for the results with the DT were also reduced, indicating that while the data is still normally distributed, it was better before the log-transformation.

5.1.2 Impact on the homoscedasticity

To test for homoscedasticity we use Levene's test for variance homogeneity [15]. The null hypothesis of the test is that the variances of two populations are the same. Levene's test does not require the data to be normally distributed. Other tests, e.g., the F-test or Bartelett's test [16] require normally distributed data and cannot be applied here (see Section 5.1.1). Table 2 shows the variances of the results and the results of Levene's test with and without the log-transformation. For both AUC and F-measure, the $p - values$ change drastically. For AUC, the $p - value$ is increased, for F-measure it is decreased. In both cases Levene's test yields the same results, i.e., the null hypothesis is still not rejected.

5.1.3 Impact on the effect size

Finally, we took a look at the impact of the log-transformation on the values of Cohen's d, which is used for the effect size correction. Table 3 shows the effect sizes with and without the log-transformation.

	AUC	F-measure
	NB vs DT	NB vs DT
No log-transformation	1.1223	0.6311
With log-transformation	1.0931	0.6488

TABLE 3
Effect sizes measured with Cohen's d.

The value of Cohen's d changes slightly. Similar to the impact on the $p - values$ of Levene's test, the changes are not consistent. For AUC, Cohen's d is slightly reduced, for F-measure the value is slightly increased.

5.2 Do the differences matter?

We use sample data to show that the different implications of the log-transformation may all lead to actual problems. Our example with real data above already shows that the log-transformation does not ensure normality. To show that it can actually break existing normality, we randomly sampled 100 instances from a normal distribution with $\mu = 0.5$, and $\sigma = 0.3$. We repeated this 100 times. Using the Shapiro-Wilk test, we determined that

- in 37 cases, the data was normally distributed with and without the log-transformation;
- in 2 cases the data was not normally distributed with and without the log-transformation;
- in 2 cases the data was normally distributed with the log-transformation, but not without the transformation; and
- in 59 cases the data was normally distributed without the log-transformation, but not with the transformation.

For this example, the log-transformation had a negative effect for 59% of the repetitions and a positive effect for only 2% of the repetitions.

To show that the differences we observed with Levene's test and with Cohen's d matter, we use the sample data shown in Table 4. To demonstrate the problem with the variance, we use $large$ and $small$. Both samples have the exact same variance, as $small$ is created by subtracting 0.3 from $large$. Consequently, the $p - value$ of Levene's test when comparing $large$ and $small$ is 1, i.e., as far as possible from rejecting the null hypothesis of variance homogeneity as possible. However, if the log-transformations of $large$ and $small$ are used instead, the $p - value$ of Levene's test is 0.0282, i.e., the null hypothesis is rejected and we determine that the variances are not equal. Thus, due to the log-transformation, this condition of ANOVA is actually broken. This can happen, because the relative distances between the values change due to the log-transformation, as we discussed in Section 4.2.

As for Cohen's d, we consider what happens between $large$ and $large2$. If we apply Cohen's d to compare $large$ and $large2$ directly, we get an effect size of 0.2007, i.e., barely non-negligible. With the log-transformation, this changes to an effect size of 0.1996, i.e., negligible. Consequently, the $large$ and $large2$ would be merged by the effect size correction with log-transformation and not merged without the log-transformation. Basically, ScottKnottESD determines them as significantly different in the log-space, but not significantly different based on their actual values. This is problematic, as the actual aim is to evaluate and compare differences between the performance metrics, not their logarithms, as they would be different performance metrics with different meanings.

Finally, we wanted to explore if the implications that the log-transformation can change the cluster analysis of Scott-Knott itself are valid. To this aim, we generated three normally distributed samples with 100 instances, where $\mu_1 = 0.5$, $\mu_2 = 0.6$, $\mu_3 = 0.625$, and $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$. We then applied the Scott-Knott test to the data with and without log-transformation. We repeated this 100 times. In two cases, the clustering with the log-transformation was different, thus with that kind of data the statistical analysis would yield wrong conclusions 2% of the time.

5.3 Further implications

All of the above is discussed in relation to performance metrics whose values are distributed in the interval $[0, 1]$. Once we leave that interval, the consequences usually get stronger. For example, the values of the performance metric Matthews Correlation Coefficient (MCC) are distributed in the interval $[-1, 1]$. Thus, a non-defined value for $log(0)$ is possible for MCC. Moreover, in the interval $[0, 1]$ the impact on the skewness of the log-transformation is rather small. Outside of that interval the effects are more drastic, as our plot of the data between $[-0.999, 1]$ in Figure 2 shows. However, we believe that our discussion above is sufficient for the conclusions we draw from our analysis.

6 Recommendations for the future use of ScottKnottESD

From our findings and subsequent discussion, our recommendation for the future use of ScottKnottESD is the following:

- Use the effect size correction to only consider clusters as different where the effect size suggests so.
- Make sure that your data fulfills the assumptions of ANOVA without any transformation.
- If the assumptions are not fulfilled, we suggest to switch to a different statistical test, where the assumptions are fulfilled.

In case the usage of ScottKnottESD is strongly desired, for example because the clear clustering of

Sample name	R command
$large$	`rep(c(0.95,0.97,0.94,0.96,0.84,0.86,0.86,0.95),40)`
$large2$	`xlarge-0.01`
$small$	`xlarge-0.3`

TABLE 4
Artificial samples to demonstrate the impact of the log-transformation. While the actual values are constructed, $large$ and $large2$ could be the results of a very well performing model, $small$ of an average model.

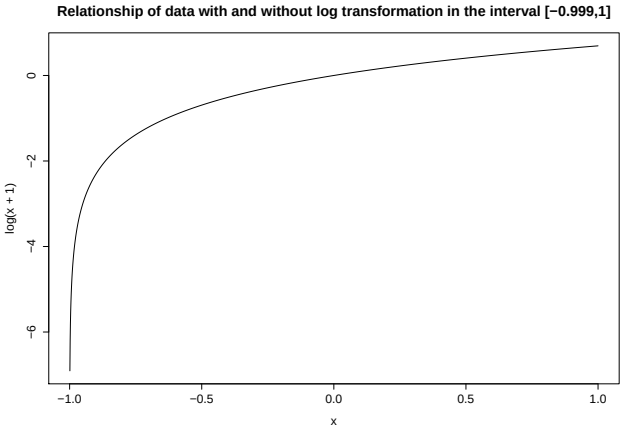

Fig. 2. Change in values due to log-transformation for MCC.

results allows good intepretations of findings, we suggest to define a meaningful transformation of the performance metric and make sure that after the transformation the assumptions are fulfilled. Most importantly, this transformation should make sense for the actual analysis that is done as part of the research performed. For example, if accuracy is transformed using the logarithm, this means that differences for low values of accuracy get a higher weigth than differences for high values of accuracy due to the skew treatment. In this case, researchers must ask if this makes sense for the intended analysis.

We would also like to note that the problem of non-normal data is very common, not only in computer science, but also in other disciplines, e.g., biomedicine and psychology [17]. Osborne [18] provides a good overview on what should be considered if data shall be transformed to achieve normality. For example, Osborne states that not only log-transformations should be considered, but also square root for counts and arcsine-root for proportions.

Finally, we conclude our recommendations with a cautionary note from Hopkins [19] regarding transformations. "*With log and other non-linear transformations, the back-transformed mean of the transformed variable will never be the same as the mean of the original raw variable. Log transformation yields the so-called geometric mean of the variable, which isn't easily interpreted. Rank transformation yields the median, or the middle value, which at least means something you can understand. The square-root and arcsine-root transformations for counts and proportions yield goodness-knows-what.*"

7 Feedback from Tantithamthavorn et al.

We send the draft of this response together with the reproducible results [11] to Tantithamthavorn *et al.* to get their feedback regarding this comment. As a result, Tantithamthavorn *et al.* double checked their results again. They found that the values in their experiments were roughly normally distributed. Moreover, the results did not change, when they did not use the log-transformation.

Tantithamthavorn *et al.* followed our recommendations for the modification of ScottKnottESD we proposed in Section 6: they removed the log-transformation and added a new function to check if the assumptions of ANOVA are met, using the same statistical tests we used in this comment. These changes resulted in version v1.2.2 of the R package which was released on May 5th, 2017 and is available archived on Zenodo [20].

8 Conclusion

Our results show that the impact of the log-transformation cannot be predicted. One cannot be sure if normality is achieved by the log-transformation. Moreover, the log-transformation may have a negative impact on various aspects of the Scott-Knott test, as well as the effect size correction of ScottKnottESD. In case assumptions of ANOVA are not met, we suggest using statistical tests where the assumptions are fulfilled, e.g., non-parametric tests like the Friedman test [21] with post-hoc Nemenyi test [22] as proposed by Demšar [23].

References

[1] C. Tantithamthavorn, S. McIntosh, A. E. Hassan, and K. a. Matsumoto, "An empirical comparison of model validation techniques for defect prediction models," *IEEE Transactions on Software Engineering*, vol. 43, no. 1, pp. 1–18, 2017.
[2] R. J. Scott and M. Knott, "A cluster analysis method for grouping means in the analysis of variance," *Biometrics*, vol. 30, 1974.
[3] R. A. Fisher, "The correlation between relatives on the supposition of mendelian inheritance," *Philosophical Transactions of the Royal Society of Edinburgh*, vol. 52, pp. 399–433, 1918.
[4] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, 1988.

[5] C. Tantithamthavorn, *ScottKnottESD: The Scott-Knott Effect Size Difference (ESD) Test*, 2016, r package version 1.1. [Online]. Available: https://CRAN.R-project.org/package=ScottKnottESD
[6] Y. Jiang, B. Cukic, and Y. Ma, "Techniques for evaluating fault prediction models," *Empirical Softw. Eng.*, vol. 13, no. 5, pp. 561–595, 2008.
[7] Y. Ma, G. Luo, X. Zeng, and A. Chen, "Transfer learning for cross-company software defect prediction," *Inf. Softw. Technology*, vol. 54, no. 3, pp. 248 – 256, 2012.
[8] J. D. Herbsleb, A. Mockus, T. A. Finholt, and R. E. Grinter, "An empirical study of global software development: Distance and speed," in *Proceedings of the 23rd International Conference on Software Engineering*, ser. ICSE '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 81–90. [Online]. Available: http://dl.acm.org/citation.cfm?id=381473.381481
[9] A. Mockus and J. D. Herbsleb, "Expertise browser: A quantitative approach to identifying expertise," in *Proceedings of the 24th International Conference on Software Engineering*, ser. ICSE '02. New York, NY, USA: ACM, 2002, pp. 503–512. [Online]. Available: http://doi.acm.org/10.1145/581339.581401
[10] B. W. Boehm, Clark, Horowitz, Brown, Reifer, Chulani, R. Madachy, and B. Steece, *Software Cost Estimation with Cocomo II with Cdrom*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2000.
[11] S. Herbold, "sherbold/replication-kit-tse-2017-comment- scottknottesd: Release of the replication kit," Mar. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.438025
[12] M. Jureczko and L. Madeyski, "Towards identifying software project clusters with regard to defect prediction," in *Proc. 6th Int. Conf. on Predictive Models in Softw. Eng. (PROMISE)*. ACM, 2010.
[13] S. Herbold, A. Trautsch, and J. Grabowski, "A comparative study to benchmark cross-project defect prediction approaches," *IEEE Transactions on Software Engineering*, vol. PP, no. 99, pp. 1–1, 2017.
[14] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.
[15] M. B. Brown and A. B. Forsythe, "Robust tests for the equality of variances," *Journal of the American Statistical Association*, vol. 69, no. 346, pp. 364–367, 1974. [Online]. Available: http://www.jstor.org/stable/2285659
[16] M. S. Bartlett, "Properties of sufficiency and statistical tests," *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 160, no. 901, pp. 268–282, 1937. [Online]. Available: http://rspa.royalsocietypublishing.org/content/160/901/268
[17] W. H. L. N. C. T. H. H. L. Y. . T. X. M. Feng, C., "Log-transformation and its implications for data analysis," *Shanghai Arch Psychiatry*, vol. 26, no. 2, pp. 105–109, 2014.
[18] J. Osborne, "Notes on the use of data transformations," *Practical Assessment, Research & Evaluation*, vol. 8, no. 6, p. 11, 2002.
[19] W. G. Hopkins, "A new view of statistics," Hopkins, W. G. (2000). A new view of statistics. Internet Society for Sport Science: http://www.sportsci.org/resource/stats/, 2000.
[20] C. K. Tantithamthavorn, "klainfo/scottknottesd: v1.2.2," May 2017.
[21] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, 1940. [Online]. Available: http://www.jstor.org/stable/2235971
[22] P. Nemenyi, "Distribution-free multiple comparison," Ph.D. dissertation, Princeton University, 1963.
[23] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248548

Steffen Herbold Dr. Steffen Herbold is a PostDoc and substitutional head of the research group Software Engineering for Distributed Systems of Prof. Jens Grabowski at the Institute of Computer Science of the Georg-August-Universität Göttingen. His research is focused on the application of data science methods and their applications in software engineering.