cmstatr: An R Package for Statistical Analysis of Composite Material Data

Stefan Kloppenborg[1]

1 Comtek Advanced Structures, Ltd.

Summary

Strength data for composite materials used in aerospace applications, such as carbon fiber and fiberglass reinforced composites, are normally analyzed using statistical methods because of the inherent variability in the constituent materials and in the processing. The design standards for civil aviation requires that the probability of structural failure due to this variability must be minimized, and to do so, the designer must use what are called "Design Values" for each material in stress analyses and ensure they exceed the actual stresses experienced by those materials in service. Design Values are set based on the one-sided lower confidence bound of the material strength. For some types of structure, the content of this lower confidence bound is 99% with a confidence level of 95%; in this case, the confidence bound is referred to as A-Basis. For some other types of structure, the content of the lower confidence bound is instead 90% with a confidence level of 95%; in this case, the confidence bound is referred to as B-Basis. The statistical methods for calculating these basis values are outlined in Composite Materials Handbook, Volume 1, Revision G, or CMH-17-1G in short (CMH-17, 2012). The use of these methods is widely accepted by industry and civil aviation regulators.

Design Values are often adjusted to account for anticipated in-service damage and other factors, however those adjustments are outside the scope of the present software package.

For a detailed discussion of the theory and applications of tolerance bounds, the reader is referred to (Meeker, Hahn, & Escobar, 2017) or (Krishnamoorthy & Mathew, 2008).

Currently, many users use MS Excel spreadsheets to perform these analyses. The MS Excel spreadsheets typically used, such as `STAT-17` (Materials Sciences Corporation, 2008), `ASAP` (Raju & Tomblin, 2008) and `CMH17-STATS` (Keshavanarayana, 2012), use password-protected `VBA` macros to perform the computations. As such, the code cannot be audited by the user. `cmstatr` is an R package that addresses this issue by implementing the same statistical analysis techniques found in CMH-17-1G in an open-source environment.

Statement of Need

The purpose of `cmstatr` is to:

- Provide a consistent user interface for computing A- and B-Basis values and performing the related diagnostic tests in the `R` programming environment
- Allow auditing of the code used to compute A- and B-Basis values, and performing the related diagnostic tests
- Enable users to automate computation workflows or to perform simulation studies

---

Implementation Goals

cmstatr aims give a consistent interface for the user. Most functions are written to work with the `tidyverse` (Wickham et al., 2019) and most functions have similar argument lists. The intent is to make the package easy to learn and use.

The implementation of `cmstatr` also aims to avoid the use of look-up tables that are prevalent in calculation spreadsheets and minimize the use of approximations. While this decision leads to increased computation time, the typically small data sets (tens to hundreds of observations) associated with composite material test data, and the speed of modern computers make this practical for interactive programming.

Example Usage

Normally, to use `cmstatr` the user will load `cmstatr` itself as well as the `tidyverse` package.

```
library(cmstatr)
library(tidyverse)
```

cmstatr contains some example data sets, which can be used to demonstrate the features of the package. One of those data sets — `carbon.fabric.2` — will be used in the following example. This data set contains results from several mechanical tests of a typical composite material, and contains the typical measurements obtained from a test lab. In the following examples, results from tension testing in the warp fiber direction (`WT`) will be used. Part of this data set is shown below.

```
carbon.fabric.2 %>%
  filter(test == "WT") %>%
  head(10)
```

```
##      test condition batch thickness nplies strength modulus failure_mode
## 1    WT        CTD     A     0.112     14   142.817   9.285          LAT
## 2    WT        CTD     A     0.113     14   135.901   9.133          LAT
## 3    WT        CTD     A     0.113     14   132.511   9.253          LAT
## 4    WT        CTD     A     0.112     14   135.586   9.150          LAB
## 5    WT        CTD     A     0.113     14   125.145   9.270          LAB
## 6    WT        CTD     A     0.113     14   135.203   9.189          LGM
## 7    WT        CTD     A     0.113     14   128.547   9.088          LAB
## 8    WT        CTD     B     0.113     14   127.709   9.199          LGM
## 9    WT        CTD     B     0.113     14   127.074   9.058          LGM
## 10   WT        CTD     B     0.114     14   126.879   9.306          LGM
```

One common task is to calculate B-Basis values. There are several statistical methods for doing so, depending on the distribution of the data. The single-point basis functions automatically perform the following diagnostic tests:

- the maximum normed residual test for outliers within a batch (CMH-17, 2012),
- the Anderson–Darling k-Sample test to check if batches are drawn from the same (unspecified) distribution (Scholz & Stephens, 1987),
- the maximum normed residual test for outliers within the data, and
- the Anderson–Darling test for a particular probability distribution (Lawless, 1982).

Assuming that the data from the warp tension (WT) tested at elevated-temperature/wet condition (ETW) follows a normal distribution, then this can be done using the function. Note that all of the functions in `cmstatr` that compute basis values default to computing tolerance bounds with a content of $p = 0.9$ and a confidence of $conf = 0.95$, or B-Basis.

```
carbon.fabric.2 %>%
  filter(test == "WT") %>%
  filter(condition == "ETW") %>%
  basis_normal(strength, batch)
```

```
## Warning: `anderson_darling_normal` failed: Anderson-Darling test rejects
## hypothesis that data is drawn from a normal distribution
```

```
##
## Call:
## basis_normal(data = ., x = strength, batch = batch)
##
## Distribution:  Normal    ( n = 18 )
## The following diagnostic tests failed:
##     `anderson_darling_normal`
## B-Basis:   ( p = 0.9 , conf = 0.95 )
## 122.9315
```

All of the various basis functions perform diagnostic tests for each of the statistical tests mentioned above. If any of the diagnostic tests failed, a warning is shown and the test failure is also recorded in the returned object (and shown in that object's `print` method). In the example above, the output shows that the Anderson–Darling test for normality (Lawless, 1982) rejects the hypothesis that the data is drawn from a normal distribution.

Two non-parametric basis calculations, based on (Guenther, 1970) and (Vangel, 1994) are also implemented in `cmstatr`. These functions perform the same diagnostic tests, but omits the Anderson–Darling test for a particular distribution.

The diagnostic test can be run directly using `cmstatr` as well. For example, the failed diagnostic test above can be run as follows:

```
carbon.fabric.2 %>%
  filter(test == "WT") %>%
  filter(condition == "ETW") %>%
  anderson_darling_normal(strength)
```

```
##
## Call:
## anderson_darling_normal(data = ., x = strength)
##
## Distribution:  Normal ( n = 18 )
## Test statistic:  A = 0.9381665
## OSL (p-value):  0.01103075  (assuming unknown parameters)
## Conclusion: Sample is not drawn from a Normal distribution ( alpha = 0.05 )
```

If the failure of a diagnostic test is decided to be acceptable, the test result can be overridden to hide the warning in the basis function output:

Kloppenborg, S., (2020). cmstatr: An R Package for Statistical Analysis of Composite Material Data. *Journal of Open Source Software*, 5(51), 2265. https://doi.org/10.21105/joss.02265

```
carbon.fabric.2 %>%
  filter(test == "WT") %>%
  filter(condition == "ETW") %>%
  basis_normal(strength, batch, override = c("anderson_darling_normal"))

##
## Call:
## basis_normal(data = ., x = strength, batch = batch, override = c("anderson_darl
##
## Distribution:   Normal      ( n = 18 )
## The following diagnostic tests were overridden:
##     `anderson_darling_normal`
## B-Basis:    ( p = 0.9 , conf = 0.95 )
## 122.9315
```

cmstatr also provides functions for calculating basis values from data pooled across multiple testing environments, as recommended by (CMH-17, 2012). There are two methods of pooling: both calculate a measure of global variation from the data in all tested environmental conditions, then they calculate a basis value for each condition using the measure of the global variation and the individual mean values of each condition. The two methods of pooling have different underlying assumptions and hence the data must pass a different set diagnostic tests for each of the two functions.

Validation and Comparison With Existing Tools

Where possible, cmstatr has been verified against the examples given in the articles in which the statistical methods were published. Unit tests have been written so that this verification is re-checked routinely to prevent unintended regressions. Agreement between cmstatr and the examples in the original articles is within the expected numeric accuracy.

cmstatr has also been verified against existing software, such as STAT-17 (Materials Sciences Corporation, 2008), ASAP (Raju & Tomblin, 2008) and CMH17-STATS (Keshavanarayana, 2012) using several example data sets. Agreement between cmstatr and the other software is generally good, but some results differ slightly, likely due to various approximations used in the software. Comparison between cmstatr and the other software is performed within various unit tests to guard against future regressions.

The tests are automatically run each time a change is made to the code of cmstatr using a continuous integration service.

Reproducibility

It is envisioned that many users of cmstatr will use it within an R Notebook or a Jupyter Notebook. It is further envisioned that this notebook will be directly converted into the statistical analysis report. If this is done, the reader of the statistical report will be able to verify all of the detailed steps used in the statistical analysis.

Acknowledgement

References

CMH-17. (2012). *Composites materials handbook, volume 1. Polymer matrix composites guideline for characterization of structural materials* (No. CMH-17-1 Revision G). SAE International.

Guenther, W. C. (1970). Determination of sample size for distribution-free tolerance limits. *The American Statistician*, *24*(1), 44–46. doi:10.2307/2682297

Keshavanarayana, S. (2012). *CMH17 statistical analysis program for B-basis & A-basis values* (No. CMH17-STATS v2011 1.1). Witchita State University.

Krishnamoorthy, K., & Mathew, T. (2008). *Statistical tolerance regions: Theory, applications, and computation*. John Wiley & Sons. doi:10.1002/9780470473900

Lawless, J. F. (1982). *Statistical models and methods for lifetime data*. John Wiley & Sons. ISBN: 0-471-08544-8

Materials Sciences Corporation. (2008). *CMH-17 statistical analysis for B-basis and A-basis values* (No. STAT-17 Rev 5). Materials Sciences Corporation.

Meeker, W., Hahn, G., & Escobar, L. (2017). *Statistical intervals: A guide for practitioners and researchers* (Second Edition.). John Wiley & Sons. ISBN: 978-0-471-68717-7

Raju, K. S., & Tomblin, J. S. (2008). *Agate statistical analysis program* (No. ASAP-2008). Wichita State University. Retrieved from https://www.niar.wichita.edu/agate/

Scholz, F. W., & Stephens, M. A. (1987). K-sample Anderson–Darling tests. *Journal of the American Statistical Association*, *82*(399), 918–924. doi:10.1080/01621459.1987.10478517

Vangel, M. (1994). One-sided nonparametric tolerance limits, *23*, 1137–1154. doi:10.1080/03610919408813222

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., et al. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. doi:10.21105/joss.01686