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

1/29/2020

Summary

A number of statistical techniques are commonly used when analyzing strength data for composite materials used in aerospace applications, such as carbon fiber and fiberglass. Currently, many users use MS Excel spreadsheets for performing this analysis. `cmstatr` is an R package that implements the statistical analysis techniques commonly used for composite material strength data.

The design standards for civil aviation require that the probability of structural failure due to material variability is minimized. To do so, the designer must select Design Values for each material and compare those to the stresses experienced by those materials. These Design Values are selected so that, with 95% confidence, the Design Value is 99% or 90% lower confidence bound of the material strength, depending on the type of structure. These one-sided tolerance bounds are referred to as A-Basis and B-Basis values, respectively. Computing these A- and B-Basis values is the main problem that `cmstatr` addresses.

A set of statistical methods are described in a publication called the Composites Materials Handbook, or CMH-17-1G (CMH-17 2012). The use of these methods is widely accepted by industry and civil aviation regulators. The methods described in CMH-17-1G are implemented in `cmstatr`.

The MS Excel spreadsheets typically used, such as `STAT-17` (Materials Sciences Corporation 2008), `ASAP` (Raju and Tomblin 2008) and `CMH-17 STATS` [INSERT CITATION], use password-protected `VBA` macros to perform the computations. As such, the code cannot be audited by the user. `cmstatr` aims to address this by providing open-source code for performing these computations.

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 lookup tables 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.

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

```
## -- Attaching packages ---------------------------------------------------------------------------- t
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------------------------------------------------------------------------------------ tidyve
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

`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 the "warp tension" (`WT`) test 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. Depending on the distribution of the data, this can be done using one of several functions. Assuming that the data from the warp tension (WT) elevated-temperature wet (ETW) strength follows a normal distribution, this can be done as follows:

```
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. If any of the diagnostic tests fail, a warning is emitted 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. The single-point basis functions perform the following 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 and Stephens 1987), the maximum normed residual test for outliers within the data, and the Anderson-Darling test for a particular distribution (Lawless 1982).

Two non-parametric basis calculations, based on (Guenther 1969) and (Vangel 1994) are implemented. These functions perform the same diagnostic tests, but skip 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
## Significance:  0.01103075  (assuming unknown parameters)
## Conclusion: Sample is not drawn from a Normal distribution (alpha =  0.05 )
```

If it is decided that the failure of the diagnostic test is acceptable, the test can be overridden to avoid a warning from being emitted by the basis function:

```
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_darling_normal"))
##
## Distribution:  Normal    ( n =  18 )
## The following diagnostic tests were overridden:
##     `anderson_darling_normal`
## B-Basis:   ( p =  0.9 , conf =  0.95 )
## 122.9315
```

`cmstatr` provides functions for calculating basis values based on data pooled across environments, as recommended by (CMH-17 2012). These functions use the variance observed in different environmental conditions in the computation, but acknowledge the different mean values under each environmental condition.

Another common statistical technique is to determine if a sample is drawn from a particular population. This is often used to determine if data from a second manufacturing site supports the same basis values. The test often recommended for this is a test considering both the mean and minimum individual value (Vangel 2002). This test has higher power than some other tests that could be used. `cmstatr` also provides functions for computing limits based on this test. For example:

```
carbon.fabric.2 %>%
  filter(test == "WT") %>%
  filter(condition == "RTD")  %>%
  equiv_mean_extremum(strength, n_sample = 8, alpha = 0.05)
```

```
##
## Call:
```

```
## equiv_mean_extremum(df_qual = ., data_qual = strength, n_sample = 8,
##     alpha = 0.05)
##
## For alpha = 0.05 and n = 8
## ( k1 = 2.700045 and k2 = 0.6789966 )
##                   Min Individual      Sample Mean
##      Thresholds:        121.4921         135.0655
```

Comparison With Existing Tools

Reproducibility

It is envisioned that many users of `cmstatr` will…

References

CMH-17. 2012. "Composites Materials Handbook, Volume 1. Polymer Matrix Composites Guideline for Characterization of Structural Materials." CMH-17-1G. SAE International.

Guenther, William. 1969. *Determination of Sample Size for Distribution-Free Tolerance Limits. Statistical Research Report, Institute of Mathematics, University of Oslo*. 1.

Lawless, J. F. 1982. *Statistical Models and Methods for Lifetime Data*. John Wiley & Sons.

Materials Sciences Corporation. 2008. "CMH-17 Statistical Analysis for B-Basis and a-Basis Values." STAT-17 Rev 5. Materials Sciences Corporation.

Raju, K.S., and J.S. Tomblin. 2008. "Agate Statistical Analysis Program." ASAP-2008. Wichita State University.

Scholz, F.W., and M.A Stephens. 1987. "K-Sample Anderson-Darling Tests." *Journal of the American Statistical Association* 82 (399): 918–24.

Vangel, Mark. 1994. "One-Sided Nonparametric Tolerance Limits" 23: 1137–54. https://doi.org/10.1080/03610919408813222.

———. 2002. "Lot Acceptance and Compliance Testing Using the Sample Mean and an Extremum." *Technometrics* 44 (3): 242–49. https://doi.org/10.1198/004017002188618428.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.