Introducing CAMIS: An open-source, community endeavor for Comparing Analysis Method Implementations in Software

Yannick Vandendijck (Johnson & Johnson, Belgium), Christina E Fillmore (GSK, UK) & Lyn Taylor (Parexel, UK)
On behalf of the PHUSE DVOST CAMIS working group

CAMIS

Background

The pharmaceutical industry has historically been limited to commercial software, such as SAS. Recently, the use of software such as R has gained momentum, due to its flexibility, far reaching capabilities, and open-source collaboration. However, there are knowledge gaps in understanding how certain statistical analyses are computed across different software.

Aim

The aim of CAMIS is to investigate and document differences and similarities between different statistical software by providing comparison, comprehensive examples and explanations. We contribute to the confidence in reliability of open-source software by understanding how analysis results can be matched perfectly or knowing the source of any discrepancies.

Method

CAMIS is a PHUSE cross-pharma project collaboration to document in a GitHub repository, the similarities and differences in software between the implementation of common statistical analyses used in medical statistics.

Demonstrative Results

Conclusions

For many statistical analyses completely matching results are found between SAS and R. Discrepancies are generally found due to differences in default methodological choices, and due to algorithmic variation.

In the transition from proprietary to open-source technology in the industry, CAMIS can serve as a guidebook to navigate this process. Knowing the reasons for differences (different methods, options, algorithms, etc.) and understanding how to mimic analysis results across software is critical to the modern statistician and subsequent regulatory submissions.

Call to action!

You have the opportunity to contribute and help our community. Interested to join? Please contact us:

Yannick Vandendijck: yvanden2@its.jnj.com
Christina Fillmore: christina.e.fillmore@gsk.com
Lyn Taylor: lyn.taylor@parexel.com

References

1. Breslow, N. E. (1974). Covariance Analysis of Censored Survival Data. Biometrics, 30, 89–99.
2. Efron, B. (1977). The Efficiency of Cox's Likelihood Function for Censored Data. Journal of the American Statistical Association, 72, 557–565.
3. Tobin, J. (1958). Estimation of Relationships for Limited Dependent Variables. Econometrica, 26 (1), 24-36.
4. Carpenter J.R., Roger J.H. & Kenward M.G. (2013). Analysis of Longitudinal Trials with Protocol Deviation: A Framework for Relevant, Accessible Assumptions, and Inference via MI. Journal of Biopharmaceutical Statistics, 23, 1352-1371.
5. Five macros: Drug Information Association (DIA) Missing Data Working Group (2012). Reference-based MI via Multivariate Normal RM (the "five macros and MIWithD"). London School of Hygiene and Tropical Medicine DIA Missing Data.

Getting different results for the same analysis depending on which software you use?

Me too! Here's the solution

<table>
<tr><th>Analysis Method</th><th>SAS</th><th>R</th><th>Similar!?</th></tr>
<tr><td>Rounding</td><td>round(2.5)
> 3

Rounding 'away from zero'</td><td>round(2.5)
> 2

Rounding 'to the even number'</td><td>Not by default!

SAS: use rounde function

R: use janitor::round_half_up()</td></tr>
<tr><td>Cox proportional hazard regression</td><td>proc phreg

Default uses Breslow[1] method for handling ties</td><td>survival::coxph()

Default uses Efron[2] method for handling ties</td><td>Not by default!

SAS: include "TIES=EFRON" in MODEL statement

R: include "ties='breslow'" in coxph() function</td></tr>
<tr><td>Logistic regression</td><td>proc logistic

Default uses effect coding for categorical variables</td><td>stats::glm()

Default uses glm coding for categorical variables</td><td>Not by default!

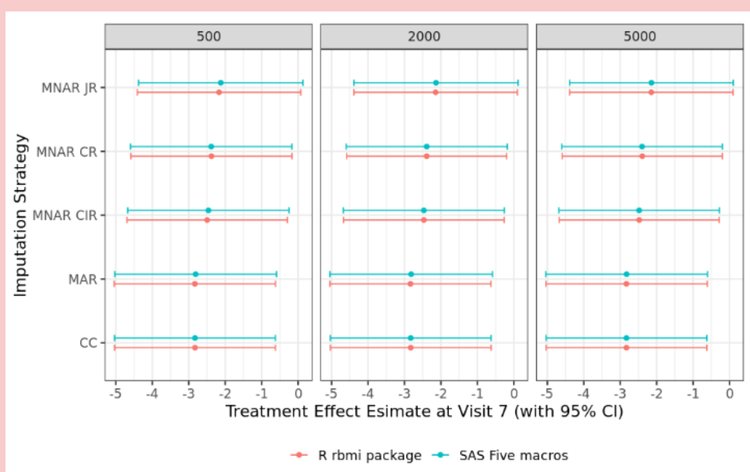
SAS: include "PARAM=GLM" or "PARAM=REFERENCE" in CLASS statement

R: No option to use effect parameterization within function</td></tr>
<tr><td>Tobit regression[3]</td><td>proc lifereg

Important to use MODEL (lower, Y), when lower is missing, then Y is used as a left-censored value</td><td>censReg::
censReg()
survival::
survreg()
VGAM::vglm()

Several R packages could be used</td><td>Yes!
<table>
<tr><th>Statistic</th><th>censReg()</th><th>survreg()</th><th>vglm()</th><th>LIFEREG</th></tr>
<tr><td>Treatment effect</td><td>1.8225</td><td>1.8225</td><td>1.8226</td><td>1.8225</td></tr>
<tr><td>Standard error</td><td>0.8061</td><td>0.8061</td><td>0.7942</td><td>0.8061</td></tr>
<tr><td>p-value</td><td>0.0238</td><td>0.0238</td><td>0.0217</td><td>0.0238</td></tr>
<tr><td>95% CI (Wald based)</td><td>0.2427 ; 3.4024</td><td>0.2427 ; 3.4024</td><td>0.2661 ; 3.3791</td><td>0.2427 ; 3.4024</td></tr>
<tr><td>σ</td><td>1.7316</td><td>1.7316</td><td>1.7317</td><td>1.7316</td></tr>
</table></td></tr>
<tr><td>Reference-based multiple imputation[4]</td><td>five macros[5]

The macros fit a Bayesian Normal RM model and then impute post-withdrawal data under a series of possible post-withdrawal profiles</td><td>rbmi package

Implements standard and reference based multiple imputation methods for continuous longitudinal endpoints</td><td>Yes!


Each panel represents a different number of imputation draws
MAR = missing at random; MNAR = missing not at random; CIR = copy increment from reference; J2R : jump to reference; CR = copy reference; CC = complete case</td></tr>
</table>

Check out more examples at