



Analysis and Validation of Oncology Clinical Trial Endpoints in Multiprogramming World

Soma Sekhar Sriadibhatla* (AstraZeneca): somasekhar.sriadibhatla@astrazeneca.com
Vikash Jain (Genpro Research): vikash.jain@genproresearch.com



CAMIS Working Group

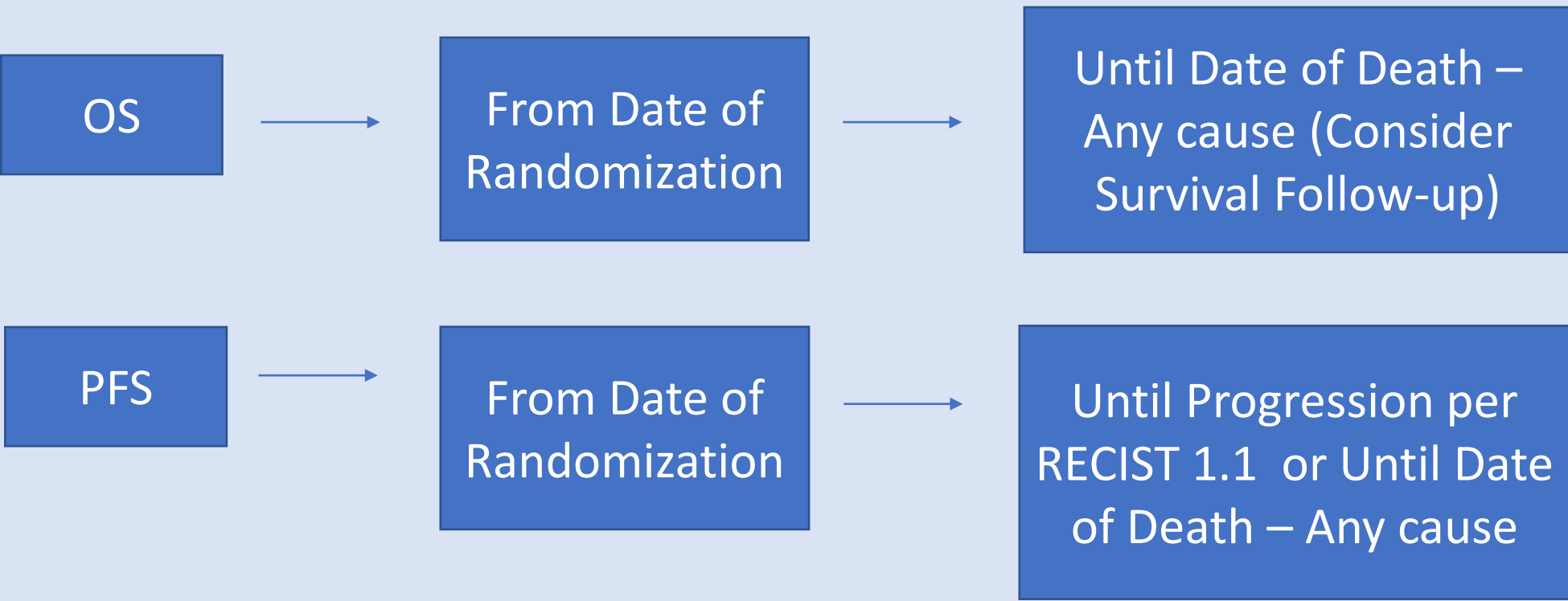


Background:

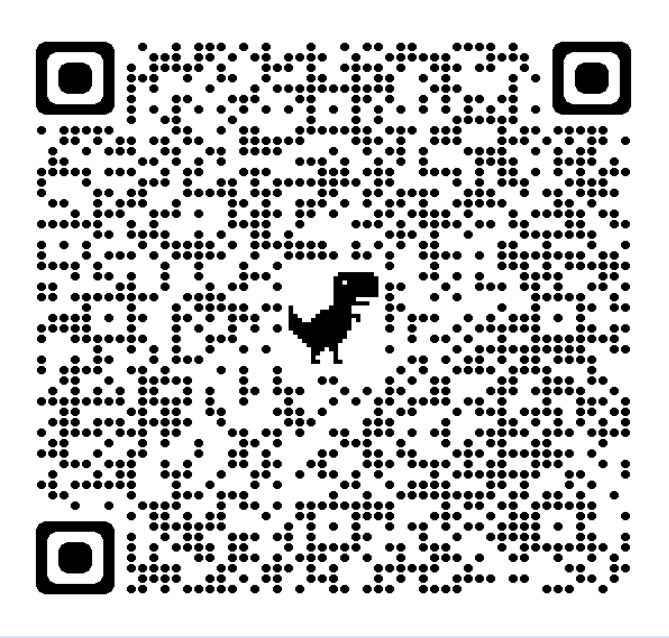
Clinical trial endpoints play a vital role in advancing research and development, providing crucial evidence for the approval of cancer drugs, as per **FDA's recommendations**. As the global nature of trials continues to expand, it is essential to ensure accurate and consistent reporting of trial endpoints across different programming languages. This poster explores the challenges, methodologies and opportunities involved in analyzing and validating oncology clinical trial endpoints in a multi programming world. This topic was developed based on Late Phase solid tumor - randomized trials and explores the possibility of automation in analysis of efficacy endpoints, and **CAMIS working group** contributions.

FDA on Oncology Clinical trial Endpoints:

FDA has standardized **Oncology trial endpoints definitions**, including study design recommendations, highlighting advantages and disadvantages across pharmaceutical industry for approvals. These definitions e.g.: **Overall Survival (OS)**, **Progression Free Survival (PFS)** are standardized to an extent that now can be considered as **“Algorithms”** of oncology efficacy programming and statistics. In many trials, these are primary efficacy endpoints and play a crucial role in approval of drugs.



Download their recommendations for your study Protocol, SAP, Dataset specifications



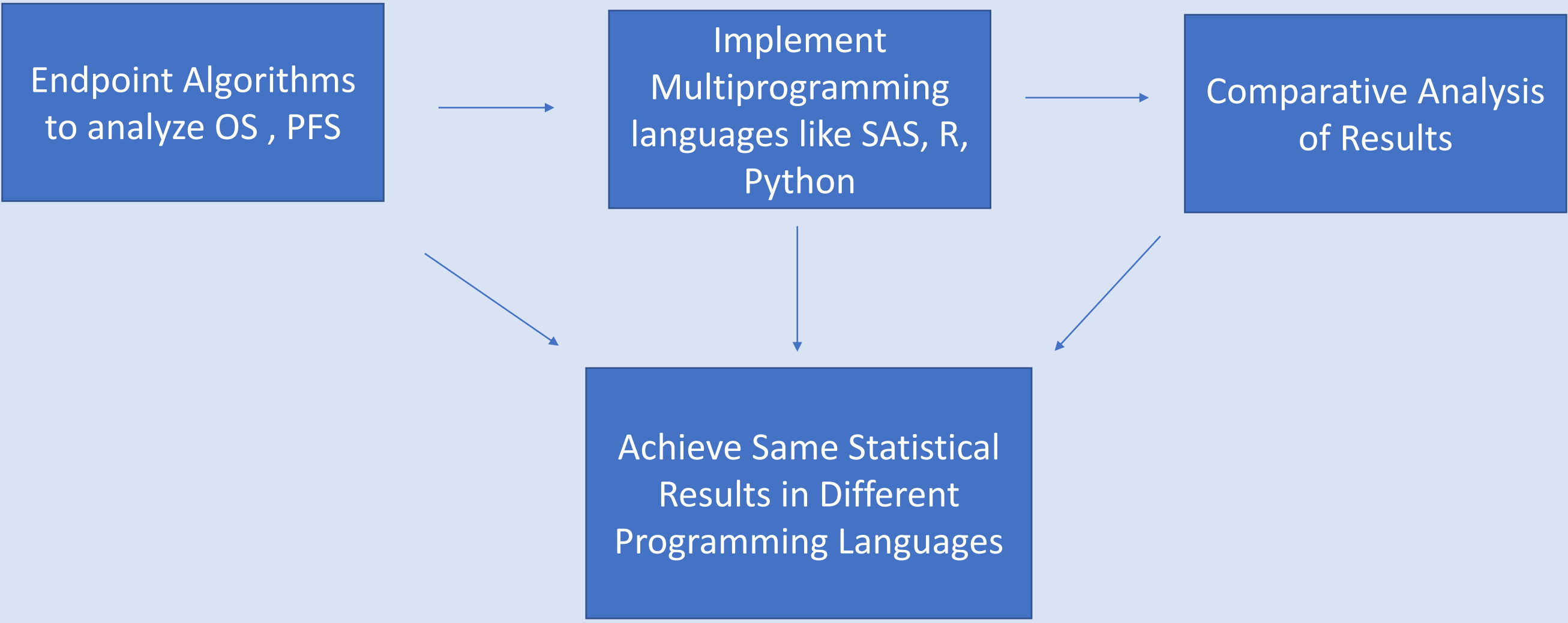
Is above too scientific? Prefer a simpler patient centric language – Scan Below!



Other than **Deaths** due to any reason on trial, including **Grade 5 Adverse events** due to drug, **Progression of Disease (PD)** becomes an important indicator to be evaluated, to perform comparison between two treatment arms, and establish statistical significance. The Progression is defined by **RECIST 1.1** (“Algorithm”), assessed by Independent Review Committee (IRC) or Investigator . The measure of interest being **Hazards ratio of OS or PFS**.

The above recommended **standardized derivations or algorithms** to analyze endpoints opens an opportunity to implement different programming languages in analysis or validation of results.

Flow Chart:



CAMIS: Comparing Analysis Method Implementations in Software

CAMIS group have been extensively working on achieving same results using different programming languages, focusing on reproducibility of statistical results. Even a few decimal points discrepancy in results can question the reliability of the programming and jeopardize the interpretation of trial data. However, we have proposed programming steps through cross collaboration between Statisticians, SAS and R programmers to achieve full replication of results. Thereby standardizing coding steps for an algorithm, Example: Log rank test, Hazards ratio of Overall Survival (OS) endpoint. With collaboration between **FDA, RECIST** - algorithms for Oncology endpoints and **CAMIS** – we can develop and document coding steps to achieve same statistical results between multiple programming languages, we can mitigate risk in multiprogramming world.

An Illustration from CAMIS - Survival Analysis of OS:

Research Statement from study SAP: Generate Hazard ratio from Cox model and confidence interval from log rank test for OS analysis on Full Analysis set or Intent-to-treat population between TRTA and TRTB – stratified by Region { Discuss using efron method } **Challenge:** A programmer will get differences in results (HR, CI) while handling the above analysis in SAS, R and Python due to **default options**, however options within the code can be used to generate desired results per specified method and type of confidence interval to be calculated

Cox Model Challenge: R uses *“efron”* by default, **SAS uses *“breslow”* by default**

Solution: R : fit.cox <- coxph(Surv(Aval, censor) ~ trt + strata(region), data = adtte) {Note - For Breslow Method specify **option: ties = “breslow”**}

SAS : proc phreg data = adtte; class trt; model aval*censor(0) = trt/rf ties = efron; strata region; run;

Programming Challenge ! **Will following code in Python generate same results as SAS and R, for efron ties, as a default?**

Py: cph.fit(adtte, duration_col = ‘Aval’, event_col = ‘censor’, strata=[‘region’])

Confidence Interval Calculation Challenge: R uses *“log”* by default, **SAS uses log-log by default** {Note - *“log-log”* prevents the problem of having confidence intervals of >1 or <0, which might happen if using *“log”* transformation}

Solution: R: fit.km<- survfit(Surv(Aval, censor) ~ trt + strata(region), conf.type = “log-log”, data = adtte)

SAS : proc lifetest data = adtte; time aval*censor(0); strata region/group = trt; run; {Note - For log transformation of CI specify **option: conftype = “log”**}

Programming Challenge! **What is default method for calculating Confidence Interval in Python?**

Py: confidence_interval = kmf.confidence_interval_()

Interested in learning how we achieved same results, R vs SAS? Visit our **Survival Analysis Github page:**



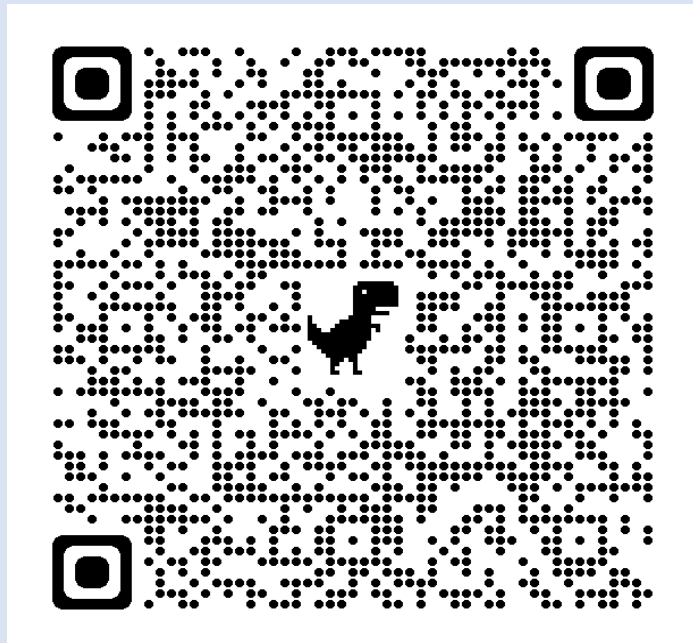
Future:

Oncology endpoint Algorithms and same Statistical results between different programming languages will provide flexibility to select languages and eliminate risk while implementing analysis and validation. Hybrid programming languages may help draw better correlation, enhanced visualization of patient data, and strict validation of results.

Innovation can be achieved in cancer endpoint analysis, through standardization of derivations – Algorithms that can generate same results in Open-Source language evolve as Machine learning models (ML).

Probably we can implement AI tools to efficiently achieve prespecified analysis of end-points, saving time, resources and bringing better care to patients. However, multiprogramming in analysis and validation is the first step.

Interested on FDA’s perception on AI/ML in Drug Development!



Acknowledgments:

CAMIS working group members, github contributors : Lyn Taylor, Christina Fillmore, Martin Brown, Michael Rimler, Mike Stackhouse, **Mia Qi**, **Min-Hua Jen**, Leon Shi, Lily Hsieh, Aiming Yang, Brian Varney, Benjamin Arancibia, Kyle Lee, Michael Kane, Jayashree Vedanayagam, Joseph Rickert, Harshal Khanolkar, Aditee Dani.

AstraZeneca-OBM, FDA Oncology Center of Excellence , RECIST Working Group

Interested in Collaboration with CAMIS-Onco group or Contribute to our **White paper?**

Contact: Soma Sekhar: somasekhar.sriadibhatla@astrazeneca.com

