phuse·virtual

Single Day Event

Activity Report of the PHUSE Japan Open-source Technology Working Group Ideas for Streamlining our Tasks via Open-source Technology

2022/12/09

Open-source Technology Working Group

phuse.global

Disclaimer

- The contents of this presentation represent the opinions of the individual presenters and do not represent the opinions or positions of the organization to which they belong.

- The contents of this presentation are omitted for convenience of explanation, and may not always contain accurate information.

Open Source Technology (OST) Working Group

Topics

- About OST WG
- OST Working Group Activities
  - Share ideas for utilizing OST
  - OST session at DIA Japan 2022

About OST WG

- **Goal**
  - To broaden the acceptance and general level of comfort with these technologies in our industry to assist in increasing their level of adoption in Japan.
- **Objective**
  - Understand industry needs and HA requirements to Open Source Technology (OST).
  - Develop general guidance including implementation, use case and possible solution for clinical study / PMS for adoption of OST.
  - Learn OST and suggest the way to implement OST in our industry.

Members

- Ippei Akiya, A2 Healthcare Corporation
- Masashi Aonuma, Astellas Pharma Inc.
- Tomoyuki Namai, Chugai Pharmaceutical Co., Ltd.
- Masashi Tanaka, EPS Corporation
- Yutaka Morioka, EPS Corporation
- Naoko Izumi, Novartis Pharma K.K.
- Yuichi Nakajima, Novartis Pharma K.K.
- Yasutaka Moriguchi, GlaxoSmithKline K.K.

OST Working Group Activities

Share ideas for utilizing OST

- Streamline our tasks
- Actual case in PK Analysis
- Actual case in ARO

Presented at Conference

- OST session at DIA Japan 2022

OST Working Group Activities

Share ideas for utilizing OST

- Streamline our tasks
- Actual case in PK Analysis
- Actual case in ARO

Note

Our purpose is to focus on value added tasks and to accelerate drug development.

Streamline Our Tasks

If the task does NOT require your brain BUT requires just your hands.

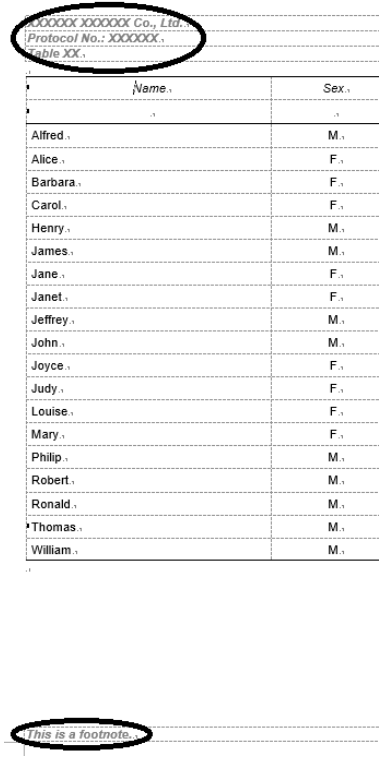
→ **OST can streamline the task!**

- Check headers and footers in TFL outputs
- Check values in Figure outputs
- Decrease the number of listings
- Develop TFL Specs by R-markdown

Streamline Our Tasks: Check headers and footers in TFL outputs

- 100-300 Tables, Figures and Listings (TFLs) in a Study. Each TFL has several headers and footers.
- Discrepancies between TFL specs and TFL outputs often happen.
- Manually checking headers and footers in TFL outputs is time-consuming.

OST can solve this problem!



XXXXXX XXXXXX Co., Ltd.
Protocol No.: XXXXXX
Table XX

Name	Sex
Alfred	M
Alice	F
Barbara	F
Carol	F
Henry	M
James	M
Jane	F
Janet	F
Jeffrey	M
John	M
Joyce	F
Judy	F
Louise	F
Mary	F
Philip	M
Robert	M
Ronald	M
Thomas	M
William	M

This is a footnote.

Streamline Our Tasks: Check headers and footers in TFL outputs

Headers/Footers in TFL outputs (.rtf) are summarized with one click!

XXXXXX XXXXXX Co., Ltd.
Protocol No.: XXXXXX
Table XX

Name	Sex
Alfred	M
Alice	F
Barbara	F
Carol	F
Henry	M
James	M
Jane	F
Janet	F
Jeffrey	M
John	M
Joyce	F
Judy	F
Louise	F
Mary	F
Philip	M
Robert	M
Ronald	M
Thomas	M
William	M

This is a footnote.

```
{\header\pard\plain\qc{
\trowd\trkeep\trqc
\cltxlrtb\clvertalt\clcbpat8\clpadt10
\clpadft3\clpadr10\clpadfr3\cellx75
39
\cltxlrtb\clvertalt\clcbpat8\clpadt10
\clpadft3\clpadr10\clpadfr3\cellx15
078
\pard\plain\intbl\b\i\sb10\sa10\ql\f
2\fs19\cf1{XXXXXX XXXXXX Co.,
Ltd.\cell}
```

RTF_headers_footers.txt - メモ帳

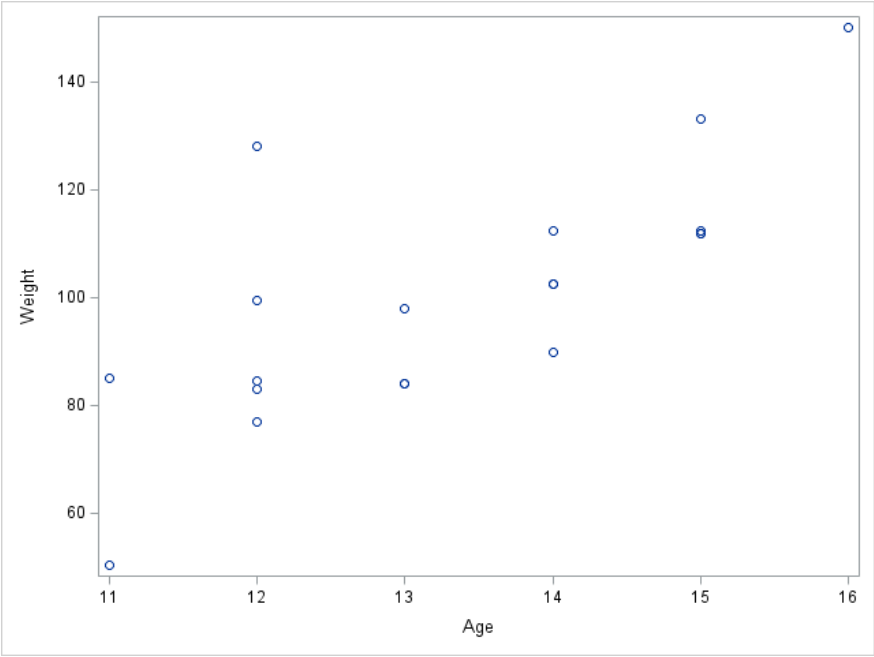
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
TLFs_Sample     header  1       XXXXXX XXXXXX Co., Ltd.
TLFs_Sample     header  2       Page {¥field{¥*¥fldinst { PAGE
TLFs_Sample     header  3       Protocol No.: XXXXXX
TLFs_Sample     header  4       Table XX
TLFs_Sample     footer  1       This is a footnote.
```

Bracket pair detection is a technical key.

```
def bracket(s, start, right):
    bra, ket = 0, 0
    for i, e in enumerate(s[start::1-2*right]):
        bra += (e == "{")*1
        ket += (e == "}")*1
        if bra == ket:
            if right == 0:
                return(s[start+1:start+i])
            elif right == 1:
                return(s[start+1-i:start])
```

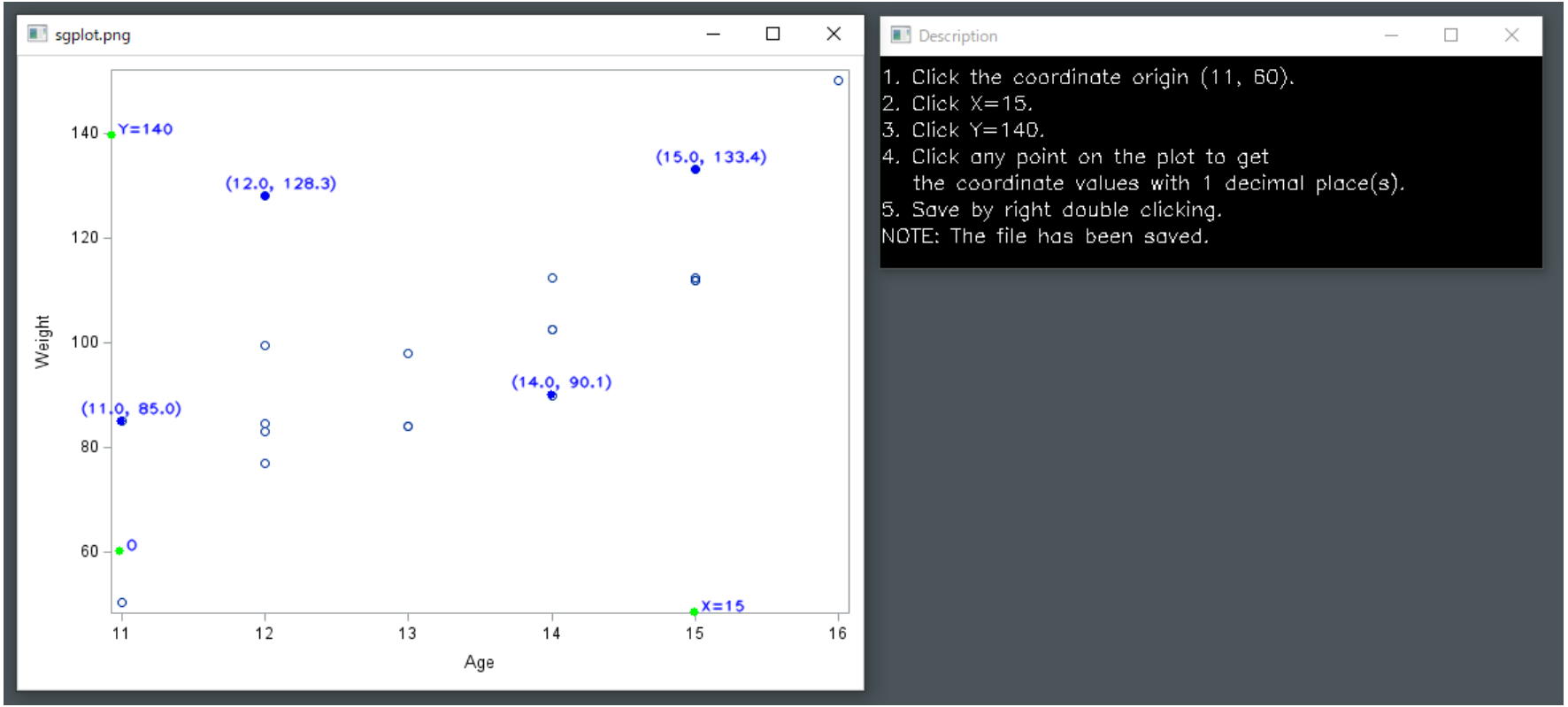
Streamline Our Tasks: Check values in Figure outputs



Manually checking each value in a figure is time-consuming

OST can solve this problem!

Streamline Our Tasks: Check values in Figure outputs

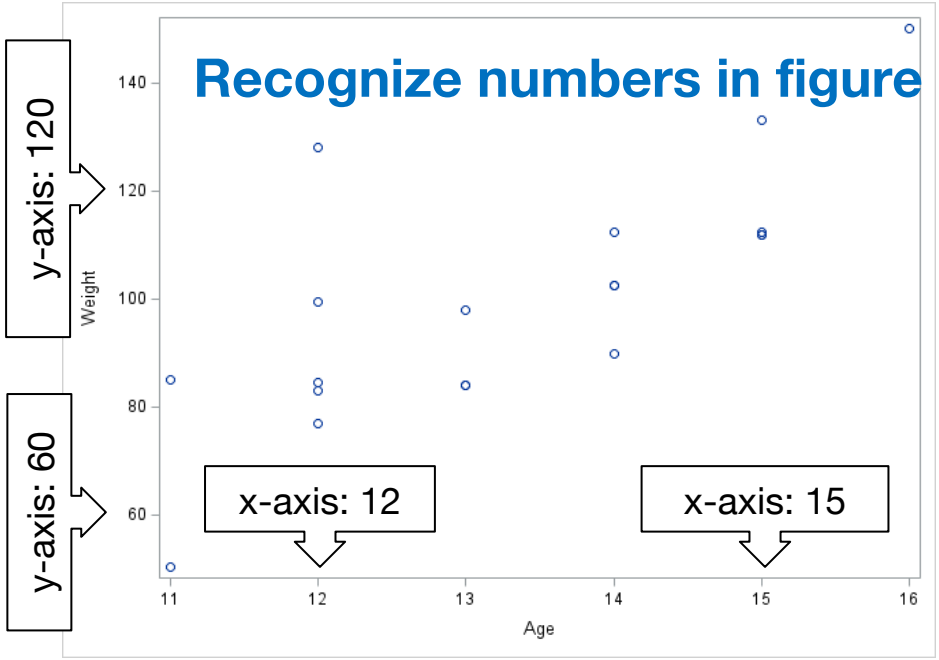


Streamline Our Tasks: Check values in Figure outputs

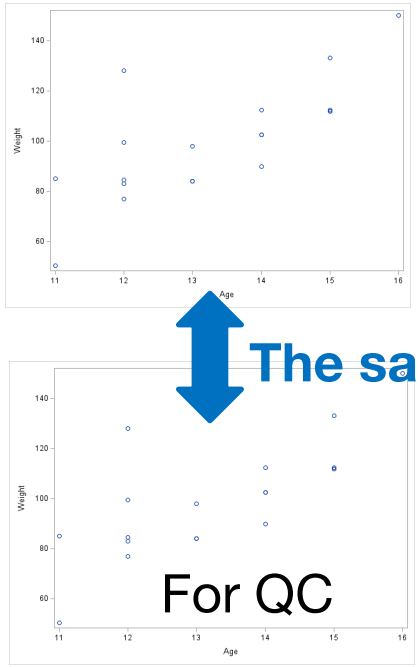
```
from pathlib import Path
import numpy as np
import cv2

path = Path("sgplot.png")
origin, x_unit, y_unit, decimal_places = (11, 60), 15, 140, 1
fonts = cv2.FONT_HERSHEY_SIMPLEX
fontp = cv2.FONT_HERSHEY_PLAIN
white, green, blue = (255, 255, 255), (0, 255, 0), (255, 0, 0)

def puttext(text, y):
    cv2.putText(description, text, (0, y), fonts, 0.5, white, 1, cv2.LINE_AA,)

description = np.zeros((160, 500, 3))
puttext(f"1. Click the coordinate origin {origin}.", 20)
puttext(f"2. Click X={x_unit}.", 40)
puttext(f"3. Click Y={y_unit}.", 60)
puttext("4. Click any point on the plot to get", 80)
puttext(f"   the coordinate values with {decimal_places} decimal place(s).", 100)
puttext("5. Save by right double clicking.", 120)


def main(event, x, y, flags, params):
    global counter, x0, y0, cx, cy

    if  event == cv2.EVENT_LBUTTONDOWN:
        if counter == 0:
            x0, y0 = x, y
            cv2.circle(img, (x, y), 3, green, -1)
            cv2.putText(img, "O", (x + 5, y), fontp, 0.8, blue, 1, cv2.LINE_AA)
            cv2.imshow(path.name, img)
            counter = 1
        elif counter == 1:
            x1 = x
            cx = (x_unit - origin[0])/(x1 - x0)
            cv2.circle(img, (x, y), 3, green, -1)
            cv2.putText(img, f"X={x_unit}", (x + 5, y), fontp, 0.8, blue, 1, cv2.LINE_AA)
            cv2.imshow(path.name, img)
            print(f"X-axis: {x_unit} eq. {x1 - x0} px.")
            counter = 2
        elif counter == 2:
            y1 = y
            cy = (y_unit - origin[1])/(y0 - y1)
            cv2.circle(img, (x, y), 3, green, -1)
            cv2.putText(img, f"Y={y_unit}", (x + 5, y), fontp, 0.8, blue, 1, cv2.LINE_AA)
            cv2.imshow(path.name, img)
            print(f"Y-axis: {y_unit} eq. {y0 - y1} px.")
            counter = 3
        elif counter == 3:
            px, py = round(origin[0] + (x - x0)*cx, decimal_places)¥
                   , round(origin[1] + (y0 - y)*cy, decimal_places)
            cv2.circle(img, (x, y), 3, blue, -1)
            cv2.putText(img, f"({px}, {py})", (x - 30, y - 5), fontp, 0.8, blue, 1, cv2.LINE_AA)
            cv2.imshow(path.name, img)
            print(px, py, sep = "¥t")
    elif event == cv2.EVENT_RBUTTONDBLCLK:
        cv2.imwrite(str(path.parent.joinpath(f"[update] {path.name}")), img,)
        puttext("NOTE: The file has been saved.", 140)
        cv2.imshow("Description", description)


counter = 0
img = cv2.imread(str(path))
cv2.imshow("Description", description)
cv2.imshow(path.name, img)
cv2.setMouseCallback(path.name, main)
cv2.waitKey(0)

# EOF[EOF]
```

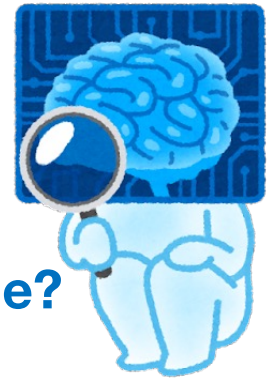
Streamline Our Tasks: Check values in Figure outputs

OpenCV is a Python library that allows us to perform image processing easily.



Compare figure with figure for QC





Streamline Our Tasks: Decrease the number of listings

Do we still need to develop voluminous listings in an era of CDISC submission?

Phuse Webinar 「Say No to Data Listings!」

- The custom that we develop all listings for all collected data started more than 20 years ago when CDISC and interactive data review tools did not exist.
- Consult with RA to clarify essential listing
- Utilize interactive data review tools such as R Shiny

Streamline Our Tasks: Develop TFL Specs by R-markdown

Current Flow

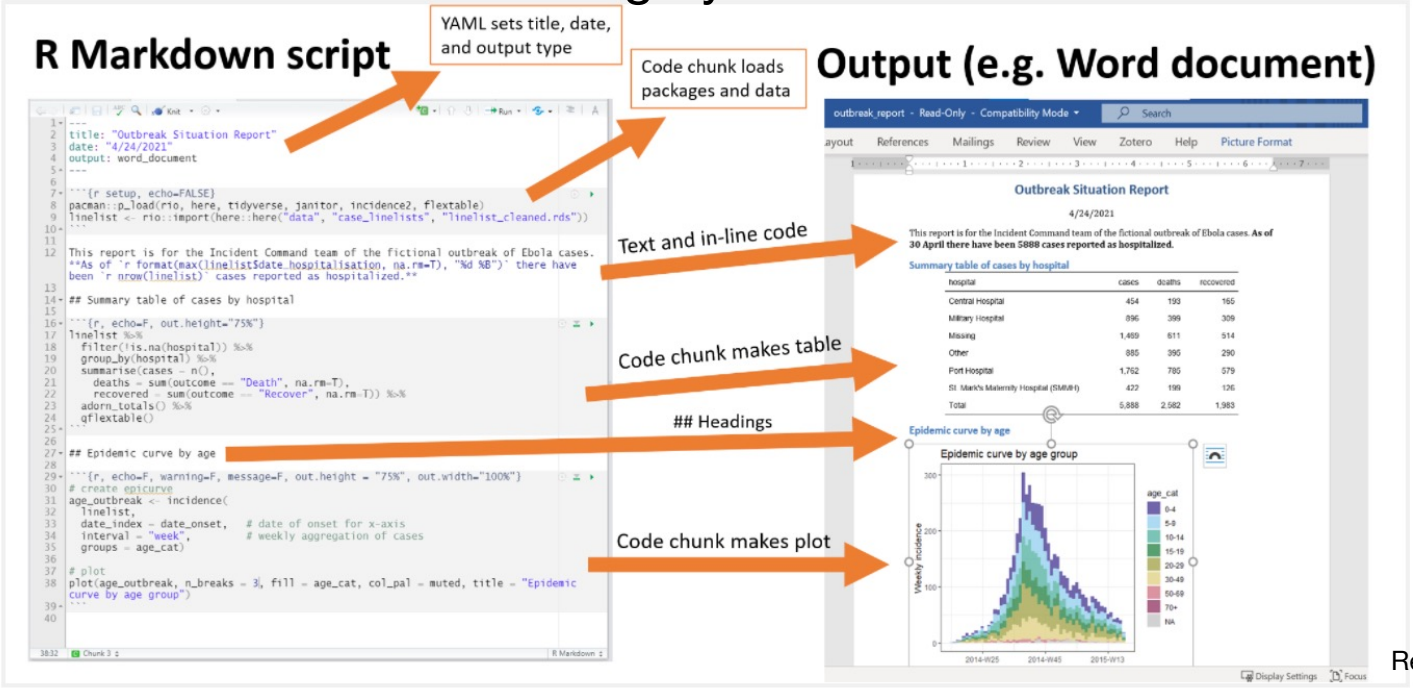
- Take long time because the next step can not start before the completion of previous step
- Need a lot of resources for fixing discrepancies between TFL specs and TFL outputs

How about R-markdown?

Streamline Our Tasks: Develop TFL Specs by R-markdown

What is R-markdown?

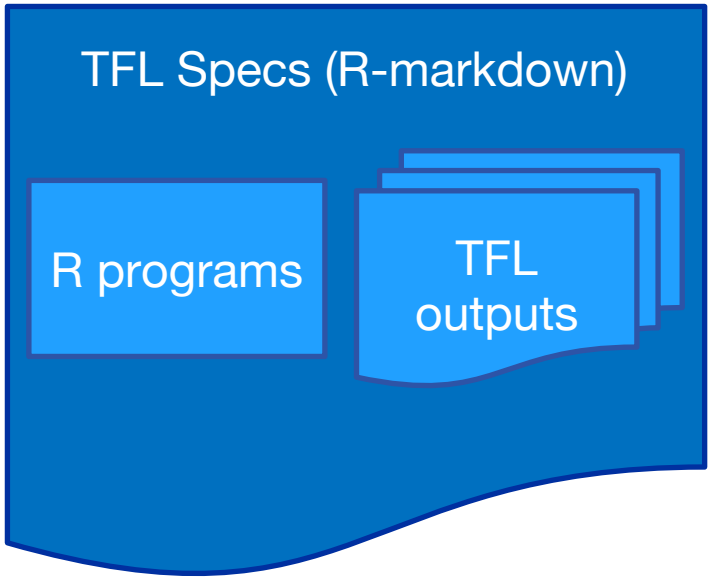
- a file format for making dynamic documents with R



Ref: [The Epidemiologist R Handbook]

Streamline Our Tasks: Develop TFL Specs by R-markdown

New Flow



- **Shorten timeline because The completion of TFL Specs = The completion of R programs = The completion of TFL outputs**
- **No discrepancy between TFL specs and TFL outputs**

OST Working Group Activities

Share ideas for utilizing OST

- Streamline our tasks
- Actual case in PK Analysis
- Actual Case in ARO

Actual case in PK Analysis

Python enables us to derive PK parameters from scratch easily!

SDTM.PC

```
pnl = PynNonlin(xy)
pnl.plot(USUBJID)
outfile("Cmax", "ng/mL", pnl.Cmax())
outfile("Tmax", "h", pnl.Tmax())
outfile("AUClast", "h*ng/mL", pnl.AUClast())
outfile("No_points_lambda_z", "", pnl.No_points_lambda_z())
outfile("Rsq_adjusted", "", pnl.Rsq_adjusted())
outfile("Lambda_z", "1/h", pnl.Lambda_z())
outfile("HL_Lambda_z", "h", pnl.HL_Lambda_z())
```

Almost the same!

Parameter	Units	% Error = (I-W)/W*100 (SD) I: Imitation W: WinNonlin®
AUClast	h*ng/mL	-0.000157275 (0.0017981)
Cmax	ng/mL	0 (0)
HL_Lambda_z	h	-0.000017272 (0.0019466)
Lambda_z	1/h	0.000017404 (0.0019468)
No_points_lambda_z		0 (0)
Rsq_adjusted		0.000148818 (0.0017010)
Tmax	h	0 (0)

Actual case in PK Analysis

R Packages for deriving PK parameters

2.1 R packages (NonCompart and ncar)

Two R packages (NonCompart and ncar) for NCA were developed in the open-source R programming language in order to allow free public use. R packages can be installed and loaded using the following scripts.

```
install.packages(c('NonCompart', 'ncar'))
library(NonCompart)
library(ncar)
```

ref: R packages for Noncompartmental Analysis (asancpt.github.io)
ref: Open-NCA – R Scripts for CDISC-based Pharmacokinetic Analysis

Actual case in PK Analysis

R Packages for deriving PK parameters

```
tblNCA(Theoph, key = "Subject", colTime = "Time", colConc = "conc", dose = 320,
       adm = "Extravascular", dur = 0, doseUnit = "mg", timeUnit = "h", concUnit = "mg/L",
       down = "Linear")
```

```
ncar::pdfNCA(fileName = "pdfNCA-Theoph.pdf", Theoph, key = "Subject",
       colTime = "Time", colConc = "conc", dose = 320, doseUnit = "mg",
       timeUnit = "h", concUnit = "mg/L", down = "Linear")
```

Actual case in PK Analysis

R Packages for deriving PK parameters

Almost the same!

Table 3.1: Comparison of NCA results generated from WinNonlin and ncar package

Parameter	WinNonlin	ncar
CMAX	7.56 mg/L	7.5600 mg/L
CMAXD	0.023625 mg/L/mg	0.0236 mg/L/mg
TMAX	2.02 h	2.0200 h
TLAG	0 h	0.0000 h
CLST	1.25 mg/L	1.2500 mg/L
TLST	24.12 h	24.1200 h
LAMZHL	8.510037883 h	8.5100 h
LAMZ	0.08145054 /h	0.0815 /h

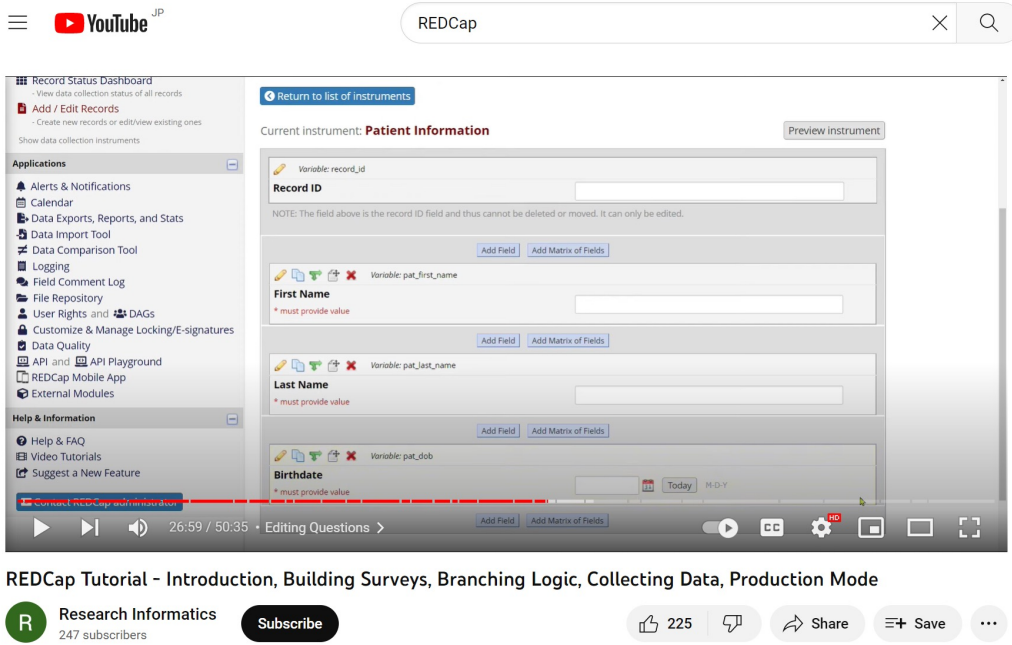
OST Working Group Activities

Share ideas for utilizing OST

- Streamline our tasks
- Actual case in PK Analysis
- Actual Case in ARO

Actual Case in ARO

REDCap (Research Electronic Data Capture) is a **FREE** EDC software for non-profit organizations



Some University in Japan has been utilizing REDCap

(Ref: FHIRを用いた臨床研究における医療情報活用)

Ref: REDCap Tutorial - Introduction, Building Surveys, Branching Logic, Collecting Data, Production Mode

* ARO: Academic Research Organization

OST Working Group Activities

Share ideas for utilizing OST

- Streamline our tasks
- Actual case in PK Analysis
- Actual Case in ARO

Presented at Conference

- OST session at DIA Japan 2022

OST session at DIA Japan 2022

“Innovative and Dynamic Statistical Analytics for Regulatory Submissions”

speaker
Modernization of Statistical Analytics: Framework and Case Studies
Isao Tsumiyama, MS
Novartis Pharma K.K.

Effective Open Source Package Management
Ippei Akiya, Msci
A2 Healthcare Corp.

Innovative and Dynamic Statistical Analysis for Regulatory Submissions
Tadeusz Lewandowski, MSc
Janssen Research & Development

Utilization of Innovative Statistical Analytics for Regulatory Submission: PMDA Perspective
Yuki Ando, PhD
Pharmaceuticals and Medical Devices Agency (PMDA)

Recap from PMDA presentation

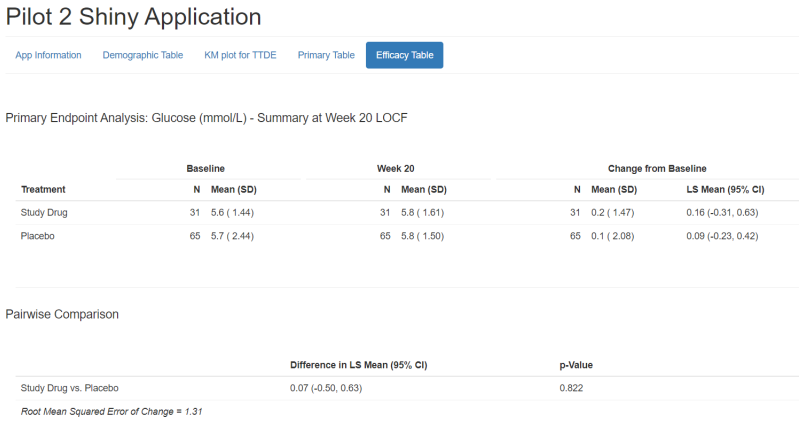
- R is being used by Stat/CP reviewers in addition to other language (SAS, JMP clinical)
- **PMDA has not expressed any restrictions on the software used for new drug applications.**
  - Refer to TCG of eData, Section 4.1.6.1 programs to be submitted.

OST session at DIA Japan 2022

- Innovative and dynamic statistical analysis for regulatory submissions
  - R consortium 2nd pilot. FDA-industry collaboration.
  - Submit a Shiny application created with the R-language for a submission package.

The R Consortium R submission Pilot 2 submission package follows the eCTD folder structure and contains the following module 5 deliverables:

- A cover letter
- SAS transport files (xpt) from CDISC ADaM/SDTM submission pilot CDISCPILOT01
- One proprietary R package "pilot2wrappers" which enables execution of the Shiny application
- An Analysis Data Reviewer's Guide (ADRG)



Reference

- Just Say No to Data Listings!
- PHUSE Safety Analytics Working Group. “Data Listings in Clinical Study Reports.” PHUSE WP-061, Version 1.0. 23 November 2021
- The Epidemiologist R Handbook
- R packages for Noncompartmental Analysis (asancpt.github.io)
- Open-NCA – R Scripts for CDISC-based Pharmacokinetic Analysis
- REDCap Tutorial - Introduction, Building Surveys, Branching Logic, Collecting Data, Production Mode
- FHIRを用いた臨床研究における医療情報活用
- R consortium 2nd pilot

Thank you for your attention!

phuse

The Global Healthcare Data Science Community

Contact Channels

UK +44 1843 609600
USA +1 609 514 5105
office@phuse.global
phuse.global

Social Media

@phusetwitta
/phusebook
/phusetube
/company/phuse