PharmaSUG 2022 - Paper DV-120

R And SAS in Analysis Data Reviewer's Guide and Data Visualization

Fan Lin, Gilead Sciences Inc.

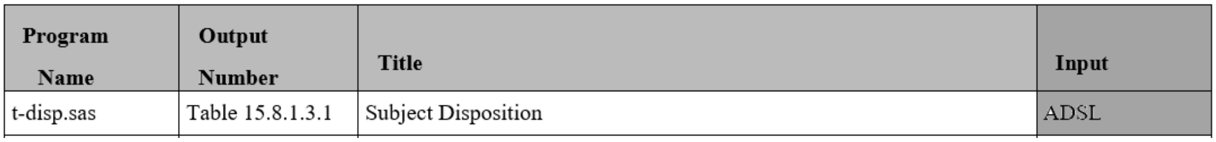
ABSTRACT

Although SAS has been the preferred programming tool in clinical studies for decades, R is gaining more popularity recently due to its flexibility and advanced graphical capabilities in data visualization.

The primary scope of this paper is to compare SAS and R in automation of Analysis Data Reviewer's Guide (ADRG) in preparation of Section 7.2, Analysis Output Programs. The secondary scope of this paper is using R for data visualization. In this paper the automation processes using SAS and R are described and the advantage or disadvantage of each language is summarized. Also the easy use of R in data visualization to create the complicated circular plot is demonstrated.

INTRODUCTION

Analysis Data Reviewer's Guide (ADRG) provides regulatory reviewers a direction to the analysis data submitted. It is an important part in item11 submission package. ADRG contains seven sections (1. Introduction, 2. Protocol Description, 3. Analysis Considerations Related to Multiple Analysis Datasets, 4. Analysis Data Creation and Processing Issues, 5. Analysis Dataset Descriptions, 6. Data Conformance Summary, 7. Submission of Programs). In the last section 7, 7.2 is Analysis Output Programs, a table including Program Name, Output Number and Title (Figure 1). Manually filling this part is very time consuming and easy to make mistakes. Programmatically generating this session can improve the accuracy and efficiency of the process. Although SAS is a reliable programming language in clinical studies and the choice by regulatory agencies it has limitations in automation of the section 7.2 in ADRG. R has demonstrated its flexibility in the usage of the packages. This paper is divided into two parts. Part one describes SAS and R in automation of ADRG section 7.2, compares SAS and R automation processes. Part two introduces the circular plot in R when SAS has challenges.



Program Name	Output Number	Title	Input
t-disp.sas	Table 15.8.1.3.1	Subject Disposition	ADSL

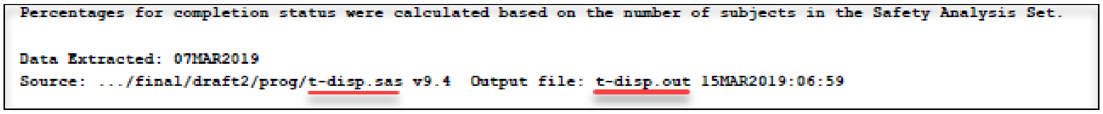
Figure 1. ADRG Section 7.2 Analysis Output Programs

PART ONE: AUTOMATION OF ADRG SECTION 7.2 USING SAS AND R

In Figure 1, ADRG section 7.2 Analysis Output Programs table, the Program Name can be found on the PDF output footnotes, "Source:", line, as shown in Figure 2, t-disp.sas. The "Source:" line on the PDF output contains the output name, t-disp.out (Figure 2). The output name is also in the Title and Footnote excel sheet (TNF), TitleKey column (Figure 3). The Output Number and Title in Figure 1 are from TNF, TFLN and Title1 columns shown in Figure 3.

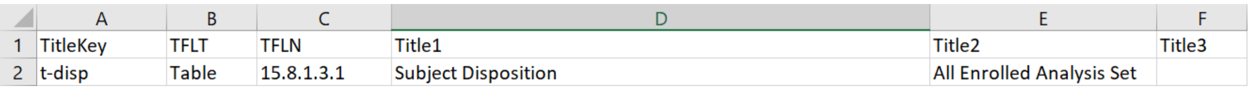
Figure 4 shows the relationships among ADRG section 7.2 table, PDF output and TNF.

The logic of the automation process is to extract Program Name and Output Name from PDF output footnote, "Source:", line, then merge with TNF by common variable, Output Name (the TitleKey on TNF), to obtain Output Number and Title, then output the merged data into rtf file for ADRG section 7.2. The next few sections will discuss the processes and codes in detail.



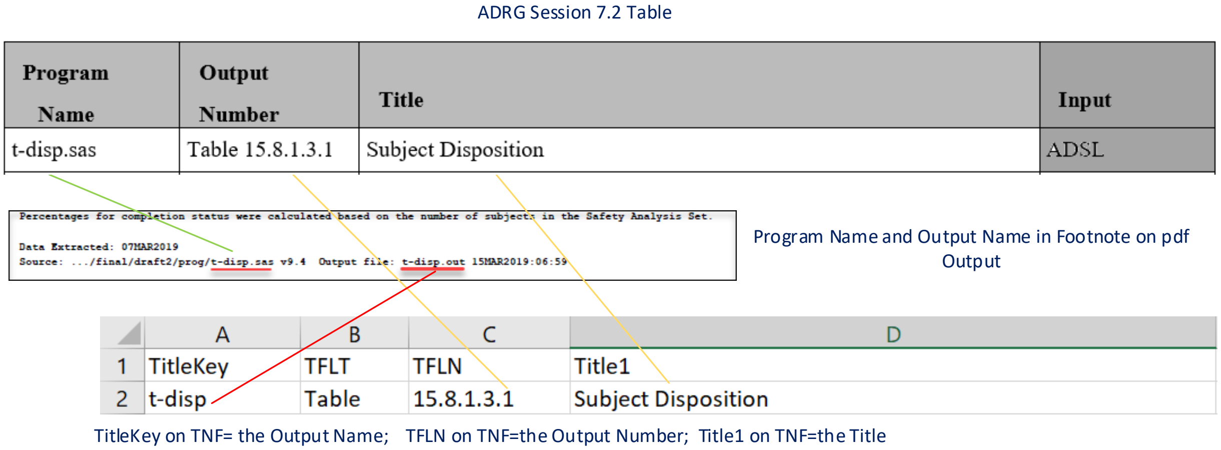
```
Percentages for completion status were calculated based on the number of subjects in the Safety Analysis Set.

Data Extracted: 07MAR2019
Source: .../final/draft2/prog/t-disp.sas v9.4  Output file: t-disp.out 15MAR2019:06:59
```

Figure 2. Footnote, “Source:”, Line on the PDF Output



	A	B	C	D	E	F
1	TitleKey	TFLT	TFLN	Title1	Title2	Title3
2	t-disp	Table	15.8.1.3.1	Subject Disposition	All Enrolled Analysis Set	

Figure 3. Output Name, Output Number, Title in TNF



ADRG Session 7.2 Table

Program Name	Output Number	Title	Input
t-disp.sas	Table 15.8.1.3.1	Subject Disposition	ADSL

```
Percentages for completion status were calculated based on the number of subjects in the Safety Analysis Set.

Data Extracted: 07MAR2019
Source: .../final/draft2/prog/t-disp.sas v9.4  Output file: t-disp.out 15MAR2019:06:59
```

Program Name and Output Name in Footnote on pdf Output

	A	B	C	D
1	TitleKey	TFLT	TFLN	Title1
2	t-disp	Table	15.8.1.3.1	Subject Disposition

TitleKey on TNF= the Output Name; TFLN on TNF=the Output Number; Title1 on TNF=the Title

Figure 4. The Relationships among ADRG Section 7.2, PDF Output And TNF

USING SAS TO AUTOMATE

1) Steps in the SAS Automation

The flow chart of the steps in SAS automation is shown on Figure 5.

1.1) Step 1: Using pipe command to create a list of all pdf files in the directory which is shown in Figure 6.

1.2) Step 2: Using do loop to infile each pdf output in the directory into sas data

1.3) Step 3a: Subsetting sas data from Step 2 for footnote “Source:” line to extract Program Name as PROGRAM and Output Name as OUTNAME, the resulting sas dataset is shown in Figure 7

1.4) Step 3b: Importing TNF into sas data, rename TitleKey as OUTNAME

1.5) Step 4: Merging the two sas datasets from Step 3a and 3b by OUTNAME, the resulting sas dataset is containing PROGNAME, OUTNAME and TITLE as shown in Figure 8

1.6) Last Step: Outputting the merged sas dataset from Step 4 into .rtf file

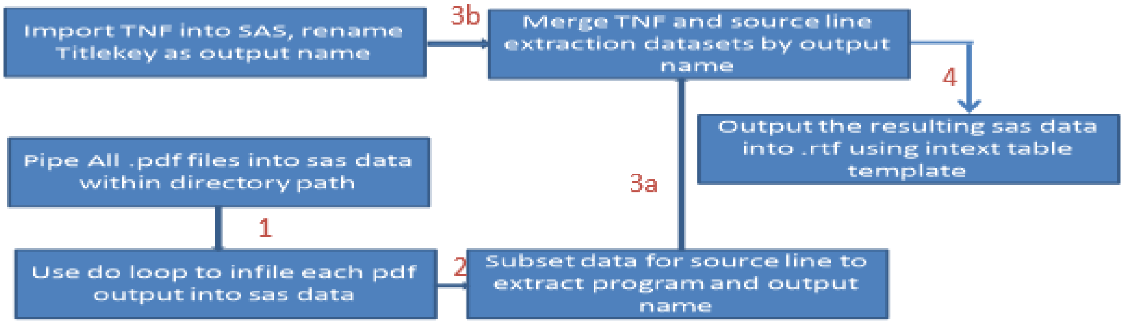
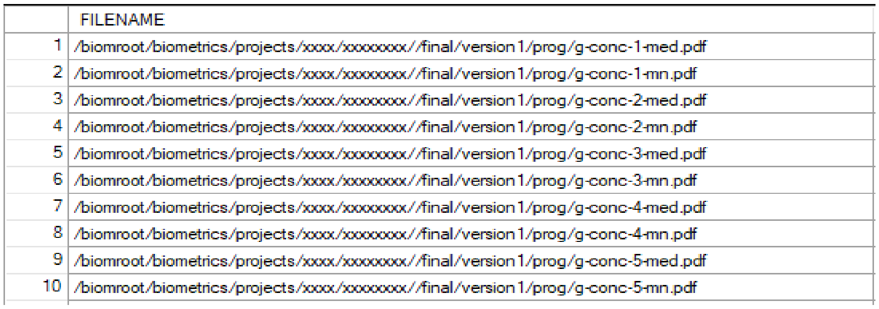
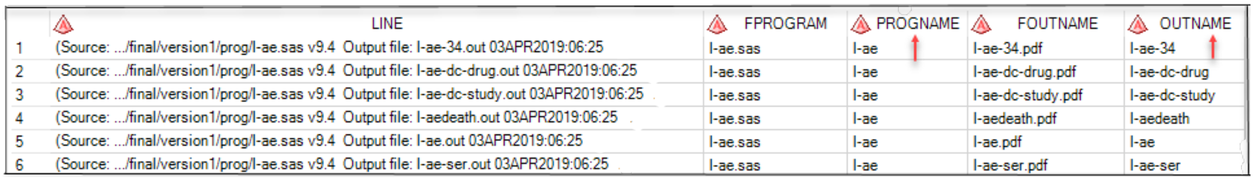

Import TNF into SAS, rename Titlekey as output name
3b
Merge TNF and source line extraction datasets by output name
4
Output the resulting sas data into .rtf using intext table template
Pipe All .pdf files into sas data within directory path
3a
1
Use do loop to infile each pdf output into sas data
2
Subset data for source line to extract program and output name

Figure 5. Flow Chart of SAS Process



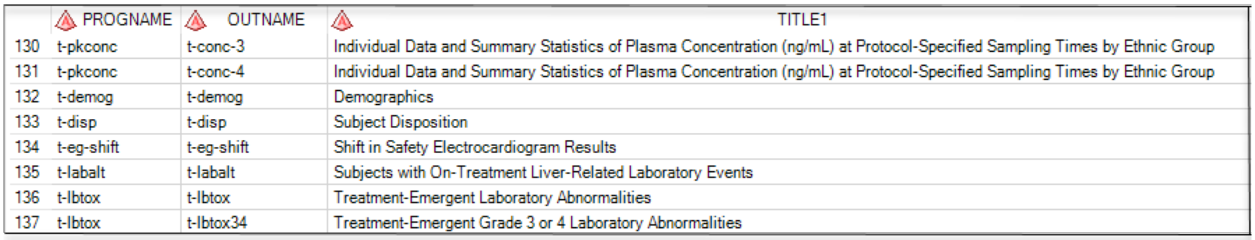
	FILENAME
1	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-1-med.pdf
2	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-1-mn.pdf
3	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-2-med.pdf
4	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-2-mn.pdf
5	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-3-med.pdf
6	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-3-mn.pdf
7	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-4-med.pdf
8	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-4-mn.pdf
9	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-5-med.pdf
10	/biomroot/biometrics/projects/xxxx/xxxxxxxx//final/version1/prog/g-conc-5-mn.pdf

Figure 6. SAS Dataset Generated from Step 1



	LINE	FPROGRAM	PROGNAME	FOUTNAME	OUTNAME
1	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-ae-34.out 03APR2019:06:25	l-ae.sas	l-ae	l-ae-34.pdf	l-ae-34
2	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-ae-dc-drug.out 03APR2019:06:25	l-ae.sas	l-ae	l-ae-dc-drug.pdf	l-ae-dc-drug
3	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-ae-dc-study.out 03APR2019:06:25	l-ae.sas	l-ae	l-ae-dc-study.pdf	l-ae-dc-study
4	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-aedeath.out 03APR2019:06:25	l-ae.sas	l-ae	l-aedeath.pdf	l-aedeath
5	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-ae.out 03APR2019:06:25	l-ae.sas	l-ae	l-ae.pdf	l-ae
6	(Source: .../final/version1/prog/l-ae.sas v9.4 Output file: l-ae-ser.out 03APR2019:06:25	l-ae.sas	l-ae	l-ae-ser.pdf	l-ae-ser

Figure 7. SAS Dataset Generated from Step 3a

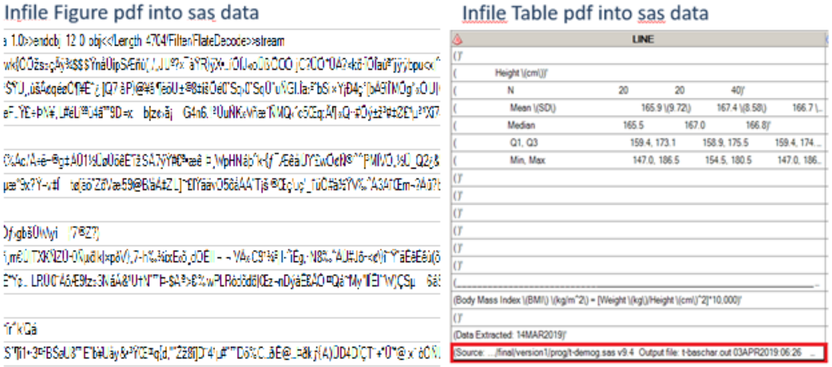


	PROGNAME	OUTNAME	TITLE1
130	t-pkconc	t-conc-3	Individual Data and Summary Statistics of Plasma Concentration (ng/mL) at Protocol-Specified Sampling Times by Ethnic Group
131	t-pkconc	t-conc-4	Individual Data and Summary Statistics of Plasma Concentration (ng/mL) at Protocol-Specified Sampling Times by Ethnic Group
132	t-demog	t-demog	Demographics
133	t-disp	t-disp	Subject Disposition
134	t-eg-shift	t-eg-shift	Shift in Safety Electrocardiogram Results
135	t-labalt	t-labalt	Subjects with On-Treatment Liver-Related Laboratory Events
136	t-lbtox	t-lbtox	Treatment-Emergent Laboratory Abnormalities
137	t-lbtox	t-lbtox34	Treatment-Emergent Grade 3 or 4 Laboratory Abnormalities

Figure 8. SAS Dataset Generated from Step 4

2) The issues in SAS automation

SAS cannot extract text from figure pdf output. On the left side of Figure 9, it shows the sas dataset from infile the figure pdf output. The Source: line in the footnote cannot be identified. On the right side of Figure 9, it is the sas dataset from table pdf output, the Source: line in footnote clearly contains program name, t-demog.sas and output name, t-baschar.out. SAS can extract expected text from table/listing pdf output, but SAS cannot extract meaningful text from figure output. In addition, SAS codes become complicated with pipe command and do loop. To solve these issues, we turn to R. The next session will introduce R approach in the automation.



Infile Figure pdf into sas data

a 1.0>>endobj 12 0 obj<</Length 4704/Filter/FlateDecode>>stream
[illegible]
[illegible]
[illegible]

[illegible]
[illegible]

)f¡gbš0Wyi '7®Z?)
[illegible]
[illegible]

Ÿr*kGá
[illegible]

Infile Table pdf into sas data

```
LINE
()'
(          Height \(cm\))'
(            N                    20          20          40)'
(            Mean \(SD\)             165.9 \(9.72\)   167.4 \(8.58\)   166.7 \...
(            Median                165.5        167.0        166.8)'
(            Q1, Q3                159.4, 173.1    158.9, 175.5    159.4, 174...
(            Min, Max              147.0, 186.5    154.5, 180.5    147.0, 186...
()'
()'
()'
()'
()'
()'
(__________________________________________________________________...
(Body Mass Index \(BMI\) \(kg/m^2\) = [Weight \(kg\)/Height \(cm\)^2]*10,000)'
()'
(Data Extracted: 14MAR2019)'
(Source: .../final/version1/prog/t-demog.sas v9.4  Output file: t-baschar.out 03APR2019:06:26  ...
```

Figure 9. SAS Datasets from Infile Figure and Table PDF Outputs

USING R TO AUTOMATE

1. Steps in the process

   The flow chart of the steps in R automation is shown on Figure 10.

   1.1) Step 1: Set R working directory to pdf files directory and load pdftools, stringr, dplyr and rtf packages.

   1.2) Step 2: list.file function creates a character vector containing the names of the files in the directory which have pattern=pdf, as shown in Figure 11.

   1.3) Step 3: Apply pdf_text function to extract text from pdf outputs into a list object. This list object not only contains the text extracted from tables and listings, but also the text extracted from figures, as shown in Figure 12.

   1.4) Step 4a: Extract program name and output name from the footnote “Source:” line on each pdf output stored in the list object and combine them into dataframe. The resulting dataframe is shown in Figure 13.

   1.5) Step 4b: Use read.csv function to read TNF into dataframe as shown in Figure 14.

   1.6) Step 5: Merge the dataframes generated from Step 4a and 4b by common column, Output Name. In R the common columns in two merged dataframes do not need to have the same name. The merge function merges two dataframes x, y with arguments, by.x= and by.y=, to reference the corresponding common column names in x and y. This is the feature which is different from SAS.

   1.7) Last Step: Output the merged dataframe to rtf file using rtf package. Figure 15 is the rtf file generated which contains Program Name, Output Number and Title.

2. R codes

   The R codes used in the process are in Figure 16. The R codes are using several packages which makes the R program concise, a few lines to complete this task and achieve what SAS has limitations to achieve.

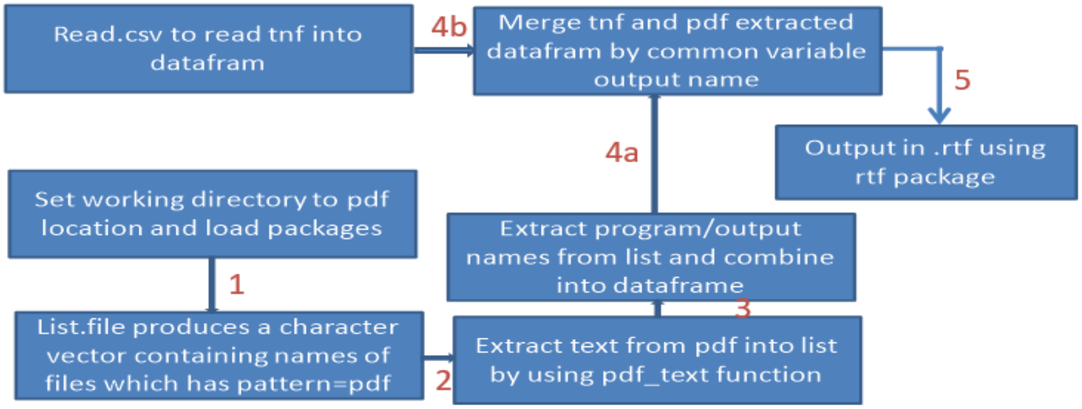
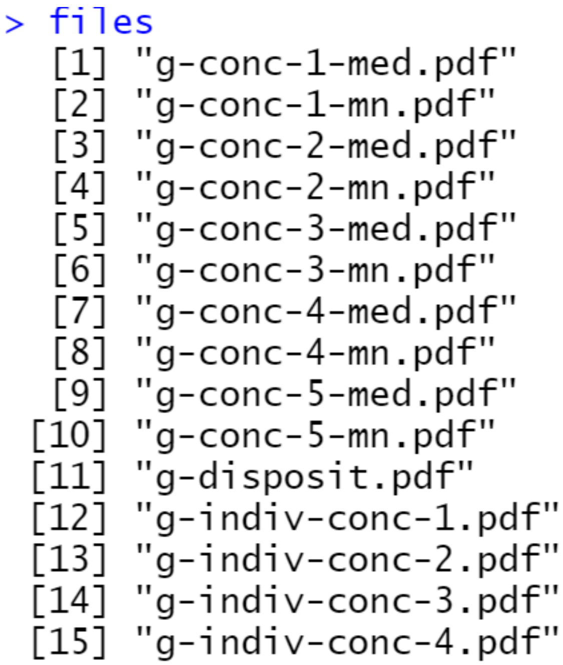
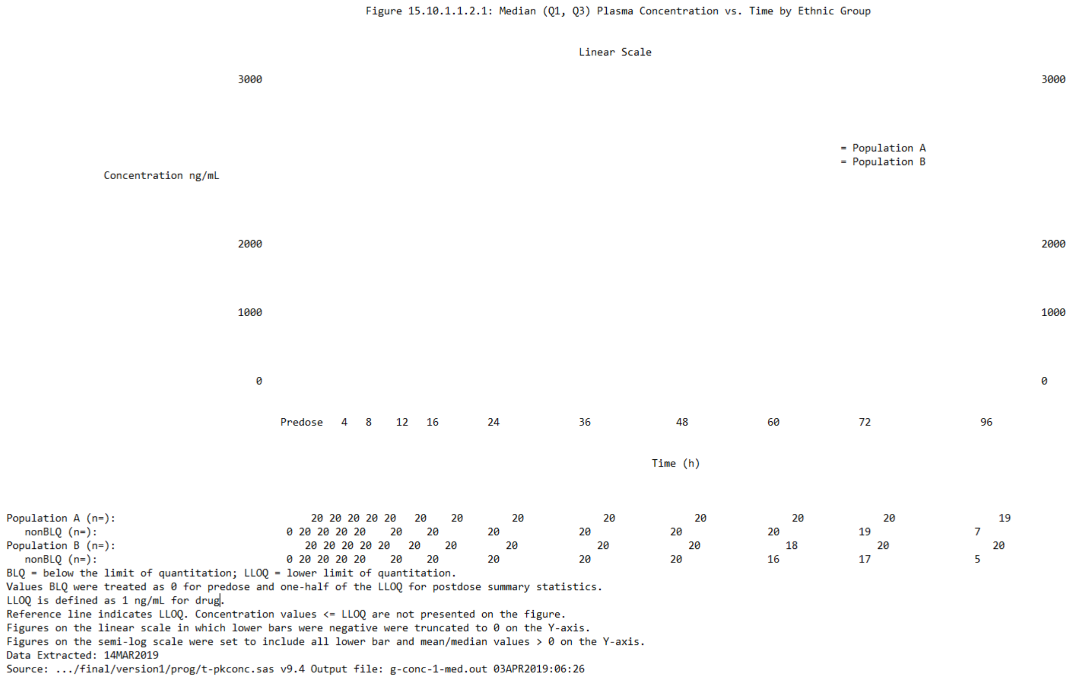

Read.csv to read tnf into dataframe
4b
Merge tnf and pdf extracted datafram by common variable output name
5
Output in .rtf using rtf package
4a
Set working directory to pdf location and load packages
1
Extract program/output names from list and combine into dataframe
3
List.file produces a character vector containing names of files which has pattern=pdf
2
Extract text from pdf into list by using pdf_text function

Figure 10. Flow Chart of R Process



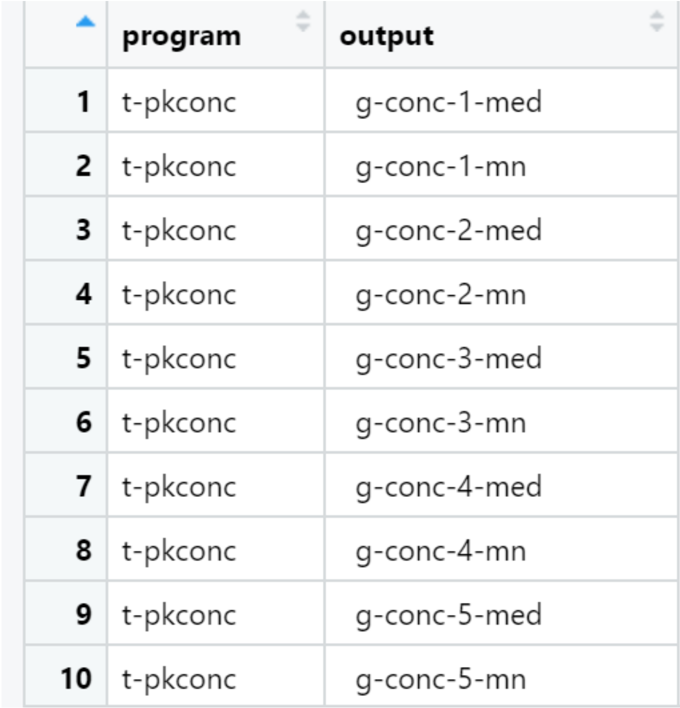
```
> files
 [1] "g-conc-1-med.pdf"
 [2] "g-conc-1-mn.pdf"
 [3] "g-conc-2-med.pdf"
 [4] "g-conc-2-mn.pdf"
 [5] "g-conc-3-med.pdf"
 [6] "g-conc-3-mn.pdf"
 [7] "g-conc-4-med.pdf"
 [8] "g-conc-4-mn.pdf"
 [9] "g-conc-5-med.pdf"
[10] "g-conc-5-mn.pdf"
[11] "g-disposit.pdf"
[12] "g-indiv-conc-1.pdf"
[13] "g-indiv-conc-2.pdf"
[14] "g-indiv-conc-3.pdf"
[15] "g-indiv-conc-4.pdf"
```

Figure 11. Vector file



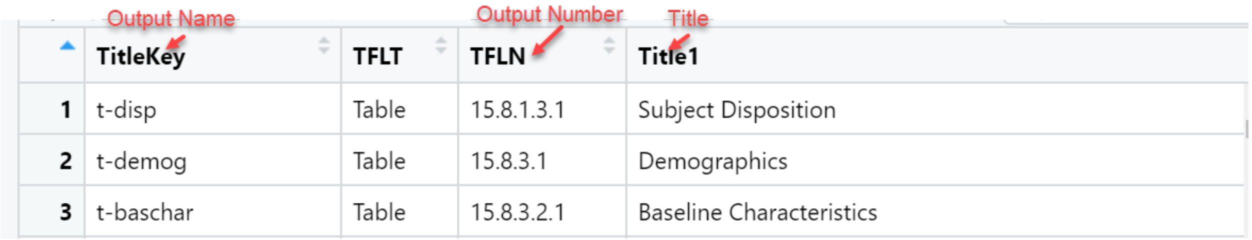
```
                                                           Figure 15.10.1.1.2.1: Median (Q1, Q3) Plasma Concentration vs. Time by Ethnic Group

                                                                                              Linear Scale
                                      3000                                                                                                                                    3000

                                                                                                                                         = Population A
                                                                                                                                         = Population B
                Concentration ng/mL


                                      2000                                                                                                                                    2000

                                      1000                                                                                                                                    1000

                                         0                                                                                                                                    0

                                             Predose   4   8    12   16        24             36              48             60             72                  96

                                                                                                          Time (h)

Population A (n=):                                20 20 20 20 20   20    20        20             20             20              20             20                 19
   nonBLQ (n=):                               0 20 20 20 20    20    20        20             20             20              20             19                 7
Population B (n=):                               20 20 20 20 20   20    20        20             20             20              18             20                 20
   nonBLQ (n=):                               0 20 20 20 20    20    20        20             20             20              16             17                 5
BLQ = below the limit of quantitation; LLOQ = lower limit of quantitation.
Values BLQ were treated as 0 for predose and one-half of the LLOQ for postdose summary statistics.
LLOQ is defined as 1 ng/mL for drug.
Reference line indicates LLOQ. Concentration values <= LLOQ are not presented on the figure.
Figures on the linear scale in which lower bars were negative were truncated to 0 on the Y-axis.
Figures on the semi-log scale were set to include all lower bar and mean/median values > 0 on the Y-axis.
Data Extracted: 14MAR2019
Source: .../final/version1/prog/t-pkconc.sas v9.4 Output file: g-conc-1-med.out 03APR2019:06:26
```

Figure 12. Contents Extracted from Figure PDF Output into the List Object



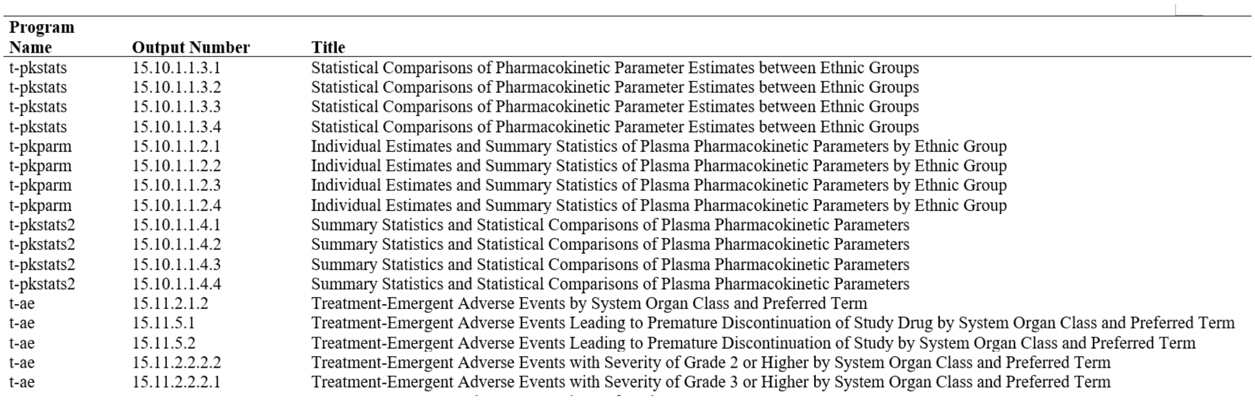
	program	output
1	t-pkconc	g-conc-1-med
2	t-pkconc	g-conc-1-mn
3	t-pkconc	g-conc-2-med
4	t-pkconc	g-conc-2-mn
5	t-pkconc	g-conc-3-med
6	t-pkconc	g-conc-3-mn
7	t-pkconc	g-conc-4-med
8	t-pkconc	g-conc-4-mn
9	t-pkconc	g-conc-5-med
10	t-pkconc	g-conc-5-mn

Figure 13. Dataframe Containing Program Name and Output Name from List Object



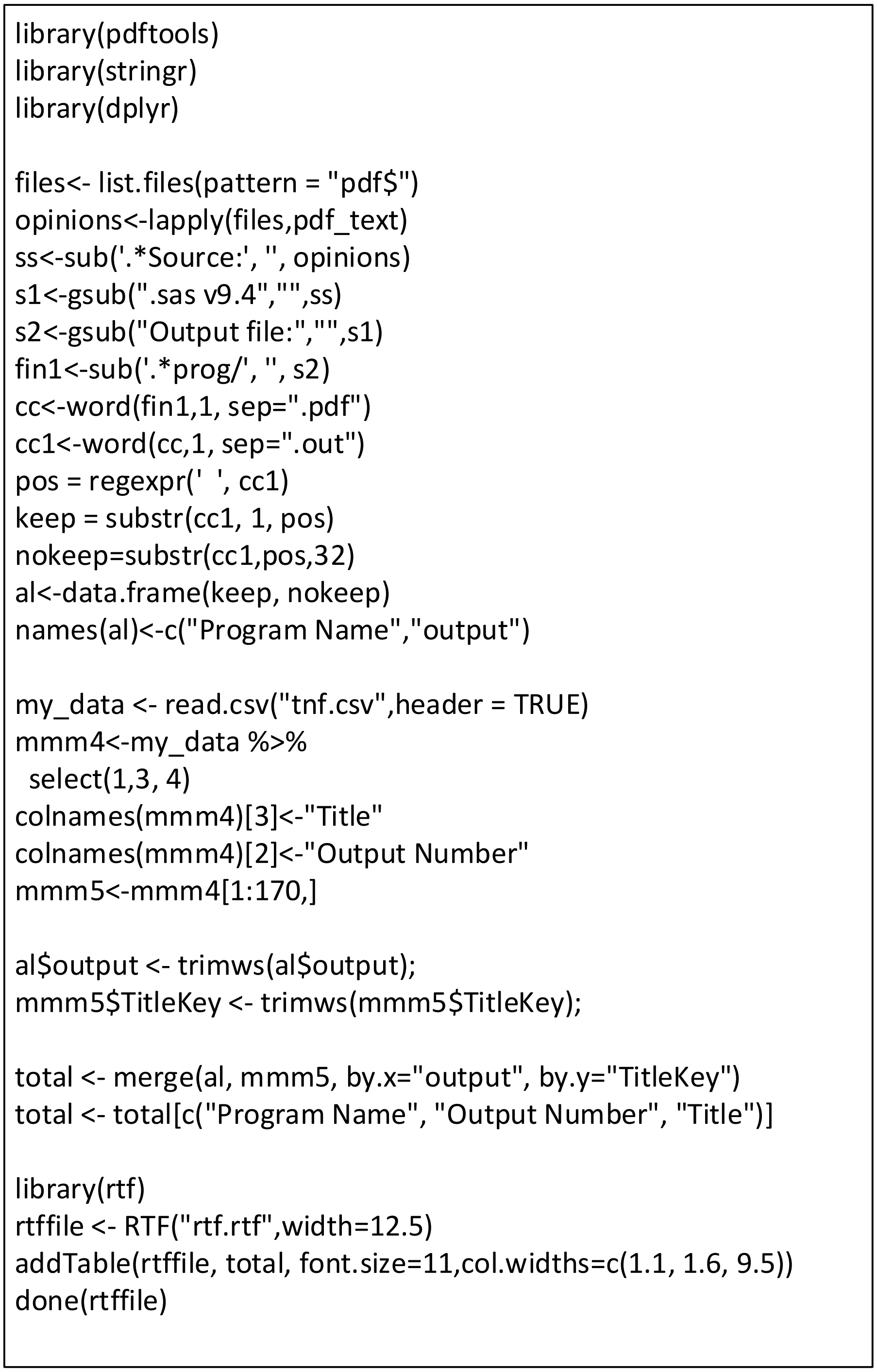
	TitleKey	TFLT	TFLN	Title1
1	t-disp	Table	15.8.1.3.1	Subject Disposition
2	t-demog	Table	15.8.3.1	Demographics
3	t-baschar	Table	15.8.3.2.1	Baseline Characteristics

Figure 14. Dataframe Created from TNF by read.csv



Program Name	Output Number	Title
t-pkstats	15.10.1.1.3.1	Statistical Comparisons of Pharmacokinetic Parameter Estimates between Ethnic Groups
t-pkstats	15.10.1.1.3.2	Statistical Comparisons of Pharmacokinetic Parameter Estimates between Ethnic Groups
t-pkstats	15.10.1.1.3.3	Statistical Comparisons of Pharmacokinetic Parameter Estimates between Ethnic Groups
t-pkstats	15.10.1.1.3.4	Statistical Comparisons of Pharmacokinetic Parameter Estimates between Ethnic Groups
t-pkparm	15.10.1.1.2.1	Individual Estimates and Summary Statistics of Plasma Pharmacokinetic Parameters by Ethnic Group
t-pkparm	15.10.1.1.2.2	Individual Estimates and Summary Statistics of Plasma Pharmacokinetic Parameters by Ethnic Group
t-pkparm	15.10.1.1.2.3	Individual Estimates and Summary Statistics of Plasma Pharmacokinetic Parameters by Ethnic Group
t-pkparm	15.10.1.1.2.4	Individual Estimates and Summary Statistics of Plasma Pharmacokinetic Parameters by Ethnic Group
t-pkstats2	15.10.1.1.4.1	Summary Statistics and Statistical Comparisons of Plasma Pharmacokinetic Parameters
t-pkstats2	15.10.1.1.4.2	Summary Statistics and Statistical Comparisons of Plasma Pharmacokinetic Parameters
t-pkstats2	15.10.1.1.4.3	Summary Statistics and Statistical Comparisons of Plasma Pharmacokinetic Parameters
t-pkstats2	15.10.1.1.4.4	Summary Statistics and Statistical Comparisons of Plasma Pharmacokinetic Parameters
t-ae	15.11.2.1.2	Treatment-Emergent Adverse Events by System Organ Class and Preferred Term
t-ae	15.11.5.1	Treatment-Emergent Adverse Events Leading to Premature Discontinuation of Study Drug by System Organ Class and Preferred Term
t-ae	15.11.5.2	Treatment-Emergent Adverse Events Leading to Premature Discontinuation of Study by System Organ Class and Preferred Term
t-ae	15.11.2.2.2.2	Treatment-Emergent Adverse Events with Severity of Grade 2 or Higher by System Organ Class and Preferred Term
t-ae	15.11.2.2.2.1	Treatment-Emergent Adverse Events with Severity of Grade 3 or Higher by System Organ Class and Preferred Term

Figure 15. RTF File Generated from rtf Package



```
library(pdftools)
library(stringr)
library(dplyr)

files<- list.files(pattern = "pdf$")
opinions<-lapply(files,pdf_text)
ss<-sub('.*Source:', '', opinions)
s1<-gsub(".sas v9.4","",ss)
s2<-gsub("Output file:","",s1)
fin1<-sub('.*prog/', '', s2)
cc<-word(fin1,1, sep=".pdf")
cc1<-word(cc,1, sep=".out")
pos = regexpr('  ', cc1)
keep = substr(cc1, 1, pos)
nokeep=substr(cc1,pos,32)
al<-data.frame(keep, nokeep)
names(al)<-c("Program Name","output")

my_data <- read.csv("tnf.csv",header = TRUE)
mmm4<-my_data %>%
  select(1,3, 4)
colnames(mmm4)[3]<-"Title"
colnames(mmm4)[2]<-"Output Number"
mmm5<-mmm4[1:170,]

al$output <- trimws(al$output);
mmm5$TitleKey <- trimws(mmm5$TitleKey);

total <- merge(al, mmm5, by.x="output", by.y="TitleKey")
total <- total[c("Program Name", "Output Number", "Title")]

library(rtf)
rtffile <- RTF("rtf.rtf",width=12.5)
addTable(rtffile, total, font.size=11,col.widths=c(1.1, 1.6, 9.5))
done(rtffile)
```

Figure 16. R Codes

PART TWO: R IN DATA VISUALIZATION

As we know, R produces better graphics for visualizing the data than SAS because R offers several graphic packages, such as ggplot2, ggplot, Lattice and circlize. SAS has challenge to provide chord diagram, a graphic method of displaying the inter-relationships between the set of data. ChordDiagram function in R plots data in a circular way, the data are arranged radically around a circle and the relationships between the data points are drawn as arcs to connect the segments. In the following Table 1, there are two dimensions, PTC as column and Treatment as row. In the treatment, there are 8 participants in Placebo and 17 in Active Study Drug. Among placebo participants, 2 are post treatment controller, six are non-controller. Post treatment controller is defined to meet the two conditions: [1] Treatment interruption (ATI) >= 24 Weeks, i.e., duration between date of ATI and treatment re-initiation needs to be >= 24 weeks. [2] HIV-1 RNA < 400 copies/mL during ATI for >= 2/3 of the time for those with ATI >= 24 weeks. Among active study drug participants, 5 are controller and 12 are non-controller. A chord Diagram is used to plot this table. R circlize package offers the convenience way for the circular plot in the complex pattern from the different dimension of data.

PTC/Treatment	Placebo n = 8	Active Study Drug n = 17	Sum
Controller	2	5	7
Non-controller	6	12	18

Table 1. Controller/Non-controller by Treatment Group

1) The steps to create circular plot from Table 1

- Load circlize package and read.csv to read in spread sheet into dataframe to contain PTC and Treatment columns
- Count the total number of controller or non-controller and concatenate the number with the value in PTC column in above dataframe, such as Controller(N=7) and Non-Controller(N=18)
- Count the total number of PLACEBO or Active Study Drug and concatenate the number with the value in treatment column in above dataframe, such as PLACEBO(N=8) and Active Study Drug (N=17)
- Use chordDiagram function to plot on columns created in above
- The generated circular plot is shown in Figure 17. The numbers on plot match in Table 1. The relationship between PTC and treatment are visualized as arcs to connect the segments on the circle, the brown arcs in Controller have two portions connecting to placebo and 5 portions connecting to active study drug, indicating there are two controllers from placebo and 5 from active study drug. The light blue arcs in Non-Controller have 6 portions connecting to placebo and 12 portions to active study drug, indicating there are 6 non-controllers in placebo and 12 in active study drug.
- The R codes for generating the circular plot are in Figure 18

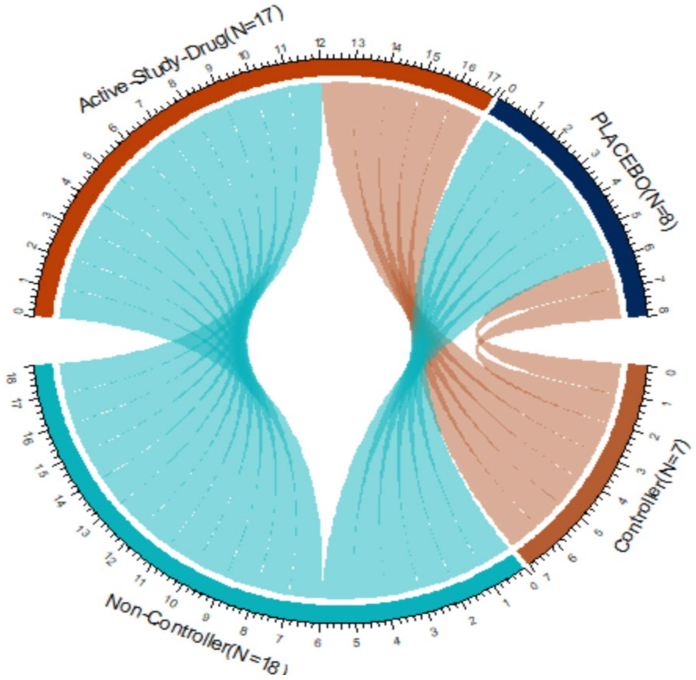
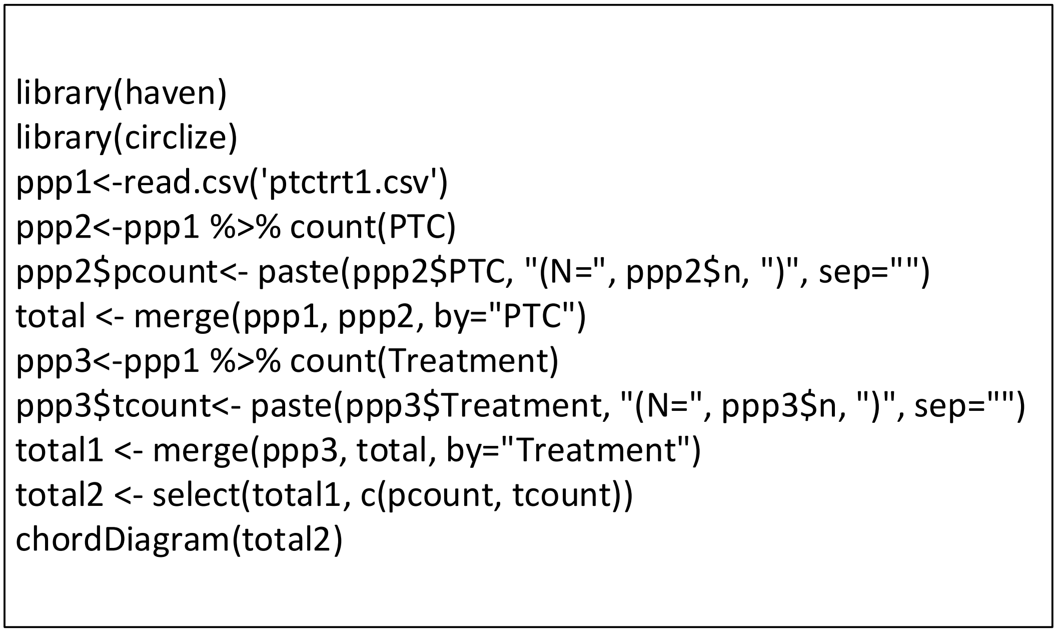


Figure 17. Circular Plot



```
library(haven)
library(circlize)
ppp1<-read.csv('ptctrt1.csv')
ppp2<-ppp1 %>% count(PTC)
ppp2$pcount<- paste(ppp2$PTC, "(N=", ppp2$n, ")", sep="")
total <- merge(ppp1, ppp2, by="PTC")
ppp3<-ppp1 %>% count(Treatment)
ppp3$tcount<- paste(ppp3$Treatment, "(N=", ppp3$n, ")", sep="")
total1 <- merge(ppp3, total, by="Treatment")
total2 <- select(total1, c(pcount, tcount))
chordDiagram(total2)
```

Figure 18. R Codes for Circular Plot

CONCLUSION

In summary, SAS is the trusted language and chosen by regulatory agencies due to its reliability. R is gaining popularity due to its flexibility in utilizing its packages. In this paper R has achieved extracting required texts from every TFL pdf output in the directory through efficiently using packages while SAS cannot extract information from figure pdf output. SAS codes are longer and needs do loops when doing the same task on TLs. In data visualization R demonstrates to have being powerful in plotting complex figure through its graphic packages provided. R is a low-level programming language. It may take longer time and more efforts to learn R comparing to SAS. However the combination of SAS and R will have no doubt to bring statistical analysis and data visualization to a new level.

REFERENCES

Gu, Z. 2014. “Circlize Implements And Enhances Circular Visualization in R.” *Bioinformatics*, DOI: 10.1093/bioinformatics/btu93.

Zhao, Shunbing. 2020. “Automating of Two Key Components in Analysis Data Reviewer’s Guide.” *Proceedings of PharmaSUG 2020 Conference*, *DV-157*. San Francisco, CA : PharmaSUG.

ACKNOWLEDGMENTS

I would like to thank my colleagues and management teams for their advice on this paper.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Fan Lin
Fan.lin@gilead.com