PharmaSUG China 2024 - Paper DS-10055

An R Shiny Application to Check Technical Conformance and Validation Criteria in NMPA Electronic Disc Submission

Junfan Zhang, MSD China, Beijing
Wang Zhang, MSD China, Beijing

ABSTRACT

Currently, as there has been a significant increase in the number of chemical and biological medication product submissions to the Center for Drug Evaluation (CDE) at the National Medical Products Administration (NMPA), the agency has progressively enhanced and strengthened the standards for submitted electronic materials. With the new "Technical Requirements for Electronic Disc Submission of Application Dossiers" ("申报资料电子光盘技术要求") came into effect, necessitating the validation of adherence to these conformance criteria prior to the release of materials. In parallel with the conformance validation software provided by the NMPA, and to facilitate a more efficient and convenient means of verifying compliance that can be widely assessed by statistical programming users, an R Shiny application has been developed specifically for this purpose. In the following sections, we will provide a detailed overview of how this R Shiny application functions to validate the criteria set forth by the NMPA.

DISCLAIMER

All views and opinions expressed in this paper represent those of the authors and should not be considered to represent advice or guidance on behalf of MSD R&D (China) Co. Ltd.

INTRODUCTION

In order to address the escalating volume of chemical and biological medication product submissions received by the CDE at NMPA, the NMPA has been consistently undertaking a process of refining and reinforcing the standards pertaining to the submitted materials. A new "Technical Requirements for Electronic Disc Submission of Application Dossiers" was released on 2023 December 11th and came into effect on 2024 March 1st.

The violation of rules from “Validation Standards for Electronic Application Dossier” (“电子申报资料验证标准”), with severity type as “error”, will lead to the rejection of the submission materials. NMPA has released an official software for the validation. However, due to the restriction of software installation and information security consideration from some companies, the NMPA software might not be widely utilized, especially by statistical programming functions for the pre-check.

As a result, to have a more efficient alternative to check the e-submission deliverables and to provide rule-followed files before formal submission, an R Shiny application is developed to validate the submission criteria.

In the first release of application, we focused on verifying the rules that are related to m5 part of the submission, the module where datasets reside hence these rules are more related to statistical programming’s delivery. We will elaborate these particular rules and validations in this paper.

VALIDATION ON SUBMISSION CRITERIA

The rules listed in the Table 1 are included and will be checked in R Shiny App v1.0. It contains nine “Error” (“错误”) and six “Prompt” (“提示信息”) with regards to the severity level of rules. For the nine ‘Error’ level rules, the requirements on file name, size, type and accessibility are defined. For the six “Prompt” level rules, file summaries and rules related to the smoothness of information obtain from PDF files are included.

No.	Description	Explanation	Severity Level
1.1	Number of Files 文件数量统计	Show the total number of all files in the current sequence. 显示当前序列中包含的所有文件的数量。	Prompt
1.2	Size of Files 文件大小统计	Show the total size of all files in the current sequence. 显示当前序列所有文件的总容量大小。	Prompt
1.4	Electronic Signature Must Meet Requirements 电子签章必须符合要求	The electronic signature of all files in the application information folder must comply with the requirements in the Technical Requirements on Compact Disc for Electronic Submission of Application Dossiers. 申请信息文件夹中所有文件的电子签章必须符合《申报资料电子光盘技术要求》的规定。	Error
2.1	Folders cannot be empty 文件夹不能为空	Empty folders (folder contains no files or sub folders) are not allowed in the directory structure. 文件夹目录结构中不允许存在空文件夹（文件夹没有文件或子文件夹）。	Error
2.2	Folders and files cannot be at the same directory level 文件夹与文件不能在同一级目录中	Folders and files should not be at the same level in the directory. 文件夹与文件不能在同一级目录中	Error
2.3	The file size should not exceed the specified maximum limit 不能超出文件大小限制	Warning information will be displayed for files that exceed the allowed size limit. Individual files have a maximum size limit of 200MB, while SAS XPT files can support up to 4GB. 超出允许大小的文件会提示警告信息。 普通单个文件最大允许 200MB，单个 SAS XPT 文件可支持 4GB。	Error
2.4	File type (file extension check) 文件类型（文件扩展名检查）	Any referenced files must have exactly one valid file extension that specified in the Technical Requirements on Compact Disc for Electronic Submission of Application Dossiers. 所有被引用文件必须有且仅有一个文件扩展名，文件扩展名必须在《申报资料电子光盘技术要求》规定的可接受的文件类型内。	Error
2.5	File and folder names must comply with the naming convention 文件和文件夹命名规范必须正确	File and folder names must comply with the naming convention specified in the Technical Requirements on Compact Disc for Electronic Submission of Application Dossiers. 文件和文件夹的命名规则必须符合《申报资料电子光盘技术要求》的规定。	Error

No.	Description	Explanation	Severity Level
2.6	The path length of folders and file names (including file extension) must comply with relevant requirements 文件夹和文件名（含扩展名）路径长度必须符合要求	The path length of folder and file name (including file extension) must comply with the requirements in the Technical Requirements on Compact Disc for Electronic Submission of Application Dossiers. 文件夹和文件名（含扩展名）路径长度必须符合《申报资料电子光盘技术要求》的规定。	Error
4.1	The PDF files must be readable. PDF 文件必须可读。	Check for damaged or unreadable PDF files (due to invalid content or zero page count) , and display an error message. 针对被破坏或不可读的 PDF 文件（因内容无效，或页码数为 0）进行检查，并提示错误信息。	Error
4.2	PDF files must not be password-protected. PDF 文件不能有密码保护	Submission of password-protected and unopenable files is not allowed. 不能提交使用密码保护且无法打开的文件。	Error
4.3	The PDF version must be correct. PDF 版本必须正确	Only PDF versions 1.4, 1.5, 1.6, 1.7, PDF/A-1, and PDF/A-2 are accepted. 允许的 PDF 文件版本为 1.4，1.5，1.6，1.7，PDF/A-1，PDF/A-2。	Prompt
4.4	PDF files with attachments are not allowed. 不允许带附件的 PDF 文件	PDF files must not contain any embedded attachments. PDF 文件中不能嵌入任何附件。	Prompt
4.7	It is recommended to create bookmarks for files with more than 5 pages. 大于 5 页的文件建议添加书签	For files with more than 5 pages, it is recommended to add bookmarks to aid navigation. 对于超过 5 页的文件，建议添加书签进行辅助导航。	Prompt
4.24	Non-standard font must be embedded in the PDF file 如使用非标准字体，需嵌入在 PDF 文件中	Standard fonts should prefered be used. If non-standard fonts are included, they must be embedded in the file. The list of standard fonts includes: "PDF 文件应尽量使用标准字体。如果包含非标准字体，则需在文件中嵌入该非标准字体。标准字体列表如下： SimSun Bold Times New Roman Times New Roman Italic Times New Roman Bold Times New Roman Bold Italic Arial Arial Italic Arial Bold	Prompt

No.	Description	Explanation	Severity Level
		Arial Bold Italic Courier New Courier New Italic Courier New Bold Courier New Bold Italic Symbol Zapf Dingbats	

Table 1. Lists of validation criteria included in R Shiny App v1.0

Shiny package is used to build the R Shiny app to validate the requirements. In this section, we will focus on how the requirements are obtained or validated.

BASIC INFORMATION

Number and Size of Files

Number and size of files are needed to be displayed to indicates the basic information of the submitted materials. These are corresponded to the 1.1 and 1.2 Indication Information from Table 1.

After specifying the directory path that requires validation, the functions list.dirs() and list.files() are utilized to retrieve all the files encompassed within the designated path. Additionally, the function file.size() is employed to ascertain the size of each individual file within the aforementioned path. The total count and cumulative size of files within the specified path are determined by aggregating the files contained within the subdirectories of the designated path.

The following code is an easy example on how the number and size of files are obtained:

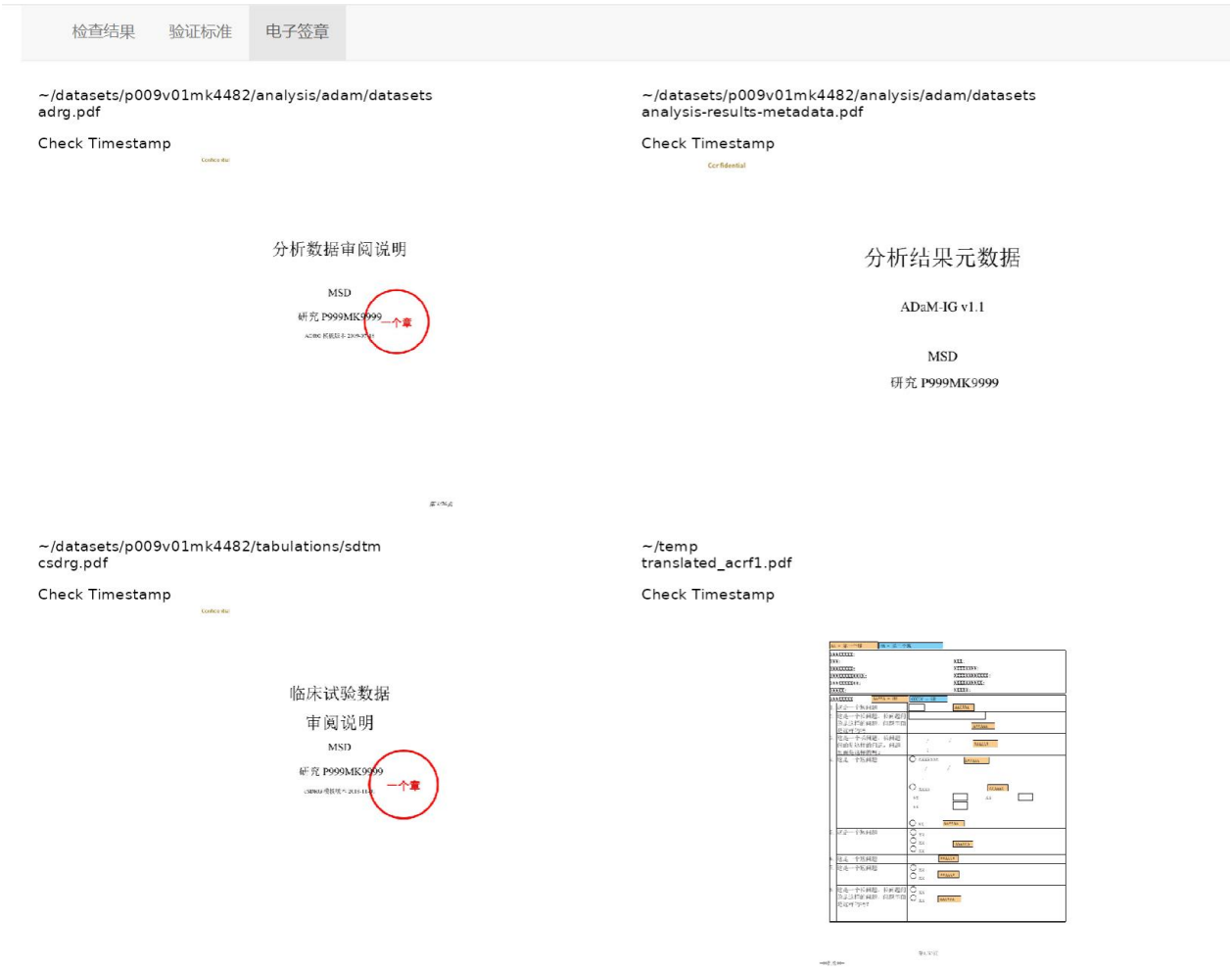
```
# Obtain path of all the files
all_folders <- list.dirs(path = file.path(folder_path()),full.names=F)

file_check1=list()
file_check2=list()
path=list()
for (i in 1:length(all_folders)) {
    file_check1[i] <- as.data.frame(list.files(path = paste(folder_path(),
all$path[i],sep='/'),all.files=T,no.. = T,full.names=T))

    file_check2[i] <- as.data.frame(list.files(path = paste(folder_path(),
all$path[i],sep='/'),all.files=T,no.. = T,full.names=F))
    }

# Display number of files and size of all documents
file_size=lapply(file_check1, file.size)
n_all_files=list()
size_all_files=list()
for (i in 1:length(file_check2)) {
    n_all_files[i]=sum(grepl("\\.",unlist(file_check2[i])))
    size_all_files[i]=sum(unlist(file_size[i])[grepl("\\.",unlist(file_chec
k2[i]))])
    }
sum_n_all_files <- sum(unlist(n_all_files))
sum_size_all_files <- paste(round(sum(unlist(size_all_files))/1073741824,
digits=2), "GB", sep=' ')
```

Electronic Signature

Regarding the electronic signature requirement, which pertains to rule 1.4, two aspects are examined:

1. Whether the electronic signature exists within the PDF file and is located on the correct page.
2. Whether the PDF file has been altered after the electronic signature was added.

As illustrated in Display 1, the first page of each PDF file is shown. If the electronic signature is not visible for a file, it indicates that there may be an issue with the existence of the electronic signature, or the signature is placed incorrectly. In such cases, the file requires updating to reflect the correct signature.



Display 1. Display of first pages of pdf files

The comparison for checking whether a PDF file has been updated after the electronic signature was added involves the modified value obtained from the pdf_info() function, which is a function within the R package "pdftools" (further details about this function will be provided in the PDF Analysis section), and the mtime value from the file.info() function in the "base" package.

Both the modified value from the pdf_info() function and the mtime value from the file.info() function represent the modification time of a file. However, the modified value from the pdf_info() function does not get updated after the electronic signature is added, while the mtime value from the file.info() function does. Since the electronic signature is added after the PDF file is finalized, the value of "modified" should always be earlier than the "mtime" value. If this rule is violated, a prompt will be displayed to remind users to check the timestamp.

FILE/FOLDER INFORMATION

The verification of folders and files contained within the designated path represents a pivotal aspect of this validation tool. It encompasses a significant portion of the rules that necessitate examination prior to submission.

6 “Error” level rule 2.1 to 2.6, from Table 1 are included in this section.

Folder Cannot be Empty

In accordance with rule 2.1, it is mandated that all folders within the designated path must contain some form of materials, be it files or subfolders. The presence of empty folders, devoid of any materials, is prohibited.

The aforementioned functions, list.dirs() and list.files(), prove to be invaluable in this context as well. By enumerating the contents of each subfolder within the specified path individually, any folder devoid of subfolders or listed files should be categorized as a folder with an error.

Folders and Files Cannot Be in the Same Directory Level

Rule 2.2 demands that the occurrence of folders and files simultaneously within any subfolder encompassed in the designated path is not acceptable.

In the output generated by the list.files() function, distinguishing between files and folders is straightforward. Files are indicated by the presence of a "." symbol in their respective entries, while folders, on the other hand, do not feature that symbol. Output 1 is a representative sample output of the list.files() function, displaying the folders, and Output 2 illustrate the example output for files.

```
[1] "datasets" "temp"
```

Output 1. Output of list.files() for folders

```
[1] "adae.xpt"   "adcm.xpt"   "adex.xpt"
[4] "adpc.xpt"   "adpp.xpt"   "adrg.pdf"
```

Output 2. Output of list.files() for files

Identifying this issue may become challenging if there are folders with the "." symbol in their names. However, it is important to note that folder names containing a "." symbol violate requirement 2.5. Hence, utilizing the previous logic of identifying files by the presence of a "." symbol and considering all entries without the symbol as folders should pose no problem.

The File Size Should Not Exceed the Specified Maximum Limit

The files included in the submitted package must adhere to the permissible size limits. Failure to comply with the specified size restrictions will result in the submission being unsuccessful. This is corresponded to rule 2.3.

The file size restrictions are presented in Table 2 for your reference.

Normal File	SAS XPT File
200MB	4GB

Table 2. File Size Limitations

The sizes of all the files have already been obtained using the file.size() function during the process of gathering the basic information of the specified path. At this point, the left task is to compare the size of each folder against the specified limitation to determine whether any files contravene the requirements.

File Type (File Extension Check)

According to rule 2.4, each file should contain only one file extension. Additionally, the file extension must fall within the specified types listed in Table 3.

Extension Type
PDF
XPT
XSL
XML
TXT

Table 3. Allowable file extensions

If a file contains two or more file extensions, it will be flagged as a violation of the requirement to have only one file extension. In such cases, the last extension will be utilized to verify compliance with the specified extension type requirement. For example, if there is a file named as “aaa.pdf.docx”, it has 2 extensions, and the extension is not in the specified types.

File and Folder Names must Comply with the Naming Convention

For rule 2.5, we examine the names of all the folders and files within the designated path. Detailed specifications for folder and file names are provided below. The names should only include:

1. Chinese characters
2. English character from “a” to “z”
3. Number from “0” to “9”
4. hyphens ("-") or underscores ("_").

Any other characters not mentioned above are not permitted in folder and file names.

Within the nomenclature of file entities, it is customary for extensions to be present, such as in the format "aaa.pdf," wherein a period character ('.') is encompassed. Consequently, prior to character assessment for a given file name, the extension (inclusive of the period) is preliminarily eliminated.

Function stri_detect_regex() from package “stringi” is used to identify whether there are characters from folder or file names, obeyed the requirement. The file and folder naming conventions must comply with the requirements specified in the " Technical Requirements for Electronic Disc Submission of Application Dossiers".

```
stri_detect_regex(string, "[^a-z0-9-_一-㐀]")
```

The pattern value ("[^a-z0-9-_一-㐀]") of this function can be explained as follow:

- `[^...]`: This is a negation character class. It matches any character that is not within the specified range or set of characters.
- `a-z`: This is a character class that represents all lowercase letters.
- `0-9`: This is also a character class that represents all digits.
- `-`: This represents a specific character, a hyphen.
- `_`: This represents a specific character, an underscore.
- `一-㐀`: This is a Unicode character range that represents Chinese characters. It includes all characters from the first Chinese character (一) to the last Chinese character (㐀) from CJK Unified Ideographs. This list included most common Chinese, Korean and Japanese Characters, if a character surpasses the designated range, it is advisable to conduct further scrutiny and examination.

Folder and File Name (Including Extension) Path Length Must Comply with Requirements

Rule 2.6 stipulates that the length of the path must satisfy the prescribed criterion. For any file within the application dossiers, the total length of the folder and file names (including the extension) starting from the root directory shall not exceed 180 characters. The length of an individual folder or file name (including the extension) shall not exceed 64 characters.

In this R Shiny application, the path of folders and files has been altered to simulate the actual path subsequent to submission. This approach enables a more meticulous evaluation of the length requirement.

As depicted in Figure 1, a sequence of characters " ZJ/2024/YPD24111111/电子新报资料 2422222222 /YPD24111111(publish)" is appended to the commencement of the path, and solely the characters subsequent to "m5" (including it) are retained. Additionally, the term "m5" has been substituted with "模块 5 临床研究报告"(Module 5, Clinical Study Report) and the term "datasets" has been substituted with "临床试验数据库"(Clinical Study Database), in accordance with the mandated structure specified by the CDE.

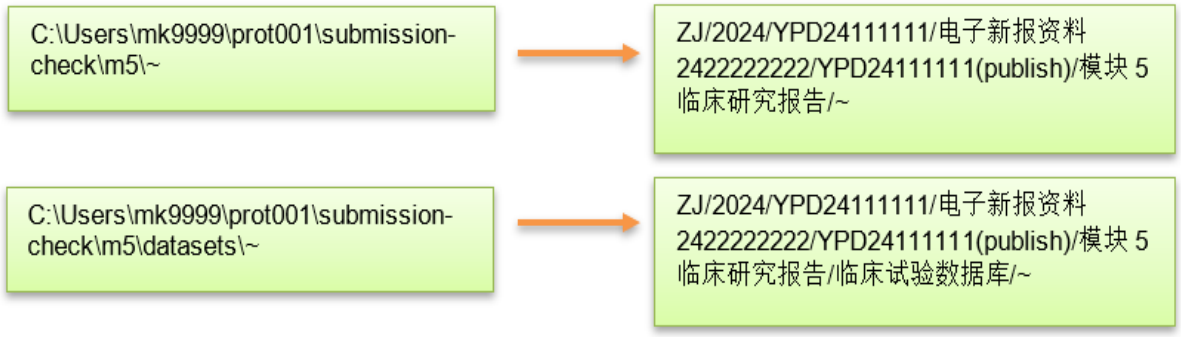


Figure 1. Output of pdf_info()

The assessment of length is performed using the revised paths.

PDF ANALYSIS

The Readability of PDF File and without Password Protection

Rule 4.1 and 4.2 helps to guarantee that user can open and obtain information smoothly from the PDF file.

R package “pdftools”, provide strongly support for checking these rules. Function pdf_info(), is mainly used here to obtain the PDF related information. In Output 3 and Output 4 the output formats of the pdf_info() function are presented, demonstrating the results for a file without password protection and a file with password protection (when no password is entered), respectively.

```
$version
[1] "1.4"

$pages
[1] 8

$encrypted
[1] FALSE

$linearized
[1] FALSE
```

```
$keys
$keys$Author
[1] "AA"

$keys$Creator
[1] "Microsoft® Word 2016"

$keys$Producer
[1] "Microsoft® Word 2016"

$created
[1] "2023-12-11 17:02:50 CST"

$modified
[1] "2023-12-11 17:02:50 CST"

$metadata
[1] "<XXXXXXXXXXXXXXXX>"

$locked
[1] FALSE

$attachments
[1] FALSE

$layout
[1] "no_layout"
```

Output 3. Output of pdf_info(), for a file without password protection

```
$encrypted
[1] FALSE

$linearized
[1] TRUE

$created
[1] "2106-02-07 01:28:15 EST"

$modified
[1] "2106-02-07 01:28:15 EST"

$locked
[1] TRUE
```

Output 4. Output of pdf_info(), for a file with password protection

When a file cannot be opened by R, which includes cases where a password is required and other situations, the "locked" value in the output of the pdf_info() function will be set to "TRUE." Furthermore, if the PDF file contains 0 page, it signifies an inherent issue with the document, rendering it unreadable. Thus, the "pages" and "locked" values from the output are analyzed to evaluate compliance with rule 4.1, determining the readability of the PDF file. If the "pages" value is 0 or the "locked" value is "TRUE," the file is identified as violating the rule.

In accordance with rule 4.2, as mentioned earlier, if the file is password-protected, the value of "lock" in the pdf_info() output should be "TRUE". To achieve more precise identification of this issue, the error

message from the pdf_text() function (included in “pdftools” package) is extracted to provide additional information. If the error message states: "Error: PDF file is locked. Invalid password?", we can conclude that rule 4.2 has not been adhered to.

Readability serves as the foundational step in the examination of all other aspects of PDF files. As a result, the subsequent rules provided in the "PDF Analysis" section will solely apply to non- locked PDF documents, disregarding those that are locked or protected.

The PDF Version must Be Correct

As stated in the definition of rule 4.3, the acceptable PDF versions are 1.4, 1.5, 1.6, 1.7, PDF/A-1, or PDF/A-2. PDF files in any other version are considered unacceptable.

As demonstrated in Output 3, the "version" value obtained from the pdf_info() output provides information about the PDF. If the "version" value is not included in the allowed list of versions, a "Prompt" will be displayed, to illustrate the violation of this rule.

PDF Files with Attachments are Not Allowed

This information can also be obtained from the output of the pdf_info() function. The "attachments" value provides valuable insight into the presence of any attachments within the PDF file. If the value of the "attachments" value in the pdf_info() output is anything other than "FALSE," it indicates the violation of rule 4.4.

It is Recommended to Create Bookmarks for Files with More than 5 Pages

Function get_bookmarks() from the R package "xmpdf" is employed to examine the presence of bookmarks within a PDF file. If a PDF file contains more than 5 pages (this information can be obtain from pdf_info() output) and the get_bookmarks() function returns no bookmarks, it indicates that the PDF file does not adhere to rule 4.7.

Non-standard Font must Be Embedded in the PDF File

The examination of PDF file fonts is conducted using the pdf_fonts() function from the "pdftools" package. An example of the output of the pdf_fonts() function is shown in Output 5.

```
   name                      type         embedded file
   <chr>                     <chr>        <lgl>    <chr>
 1 TimesNewRomanPSMT         cid_truetype TRUE     ""
 2 BCDEEE+SimSun             cid_truetype TRUE     ""
 3 TimesNewRomanPS-ItalicMT truetype     FALSE    "C:\\WINDOWS\\Fonts\\timesi.ttf"
 4 BCDFEE+Calibri            truetype     TRUE     ""
```

Output 5. Output of pdf_fonts() function

For a PDF file, if there exist one or more fonts that the name is not included in the acceptable font list and is not embedded (the embedded value is “FALSE”), the “Prompt” information will be displayed to indicate the violation of rule 4.24.

USER INTERFACE DESIGN

In this section, the UI (user interface) of the R Shiny App is displayed. Here in Display 2 is the UI of the R Shiny App.

Display 2. UI of R Shiny App

In the left panel, there are 2 inputs needed for the validation:

1. The folder path in which data package needs to be validated.
2. The folder path where users want to output additional information if errors were found.

Additionally, there are two buttons available the left panel for display the validation output:

1. "Output Outcome" ("输出检查结果") Button: When the user clicks the "Output Outcome" button, it will display the basic information of the specified path and the validation output for all the previously mentioned rules, excluding the "Electronic Signature" rule.
2. "Check Electronic Signatures" ("检查电子签章") Button: Clicking the "Check Electronic Signatures" button will display the "Electronic Signature" related information in the "Electronic Signatures" ("电子签章") tab.

These buttons allow the user to view the validation results and relevant information based on the specific rule being checked.

In the output panel, there are 3 tabs that can be used to check different output information:

1. In “Validation Outcome” (“检查结果”) tab, basic information and whether errors (except rule 1.4) happened can be confirmed. If an error happened, relative cell is highlighted in yellow with an outcome suggests the specific error ("发生此错误”).
3. In “Validation Criteria and Simulated Paths” (“验证标准与模拟路径”) tab, validation rules and their specific explanation is displayed. If the specified folder path violated any of the requirement, the rules can be easily found in this tab. Additionally, the simulated paths after submission (to validate the pathname length rule), is also displayed in this tab.
2. In “Electronic Signatures” (“电子签章”) tab, the first page of all the PDF files in the designated path is displayed. User can directly check whether the electronic signature is missed in any PDF documents.

Here is the code to implement the UI:

```
library(shiny)
library(DT)

# Define UI for application that draws a histogram
```

```
shinyUI(fluidPage(
  tags$head(
    tags$style(
      HTML("
      .my-text {
        font-family: Arial, sans-serif;
        font-size: 18px;
        font-weight: bold;
        color: black;
        margin-top: 25px;
        margin-bottom: 5px;
      }
      ")
    )
  ),

  tags$head(
    tags$style(
      HTML("
        h3 {
          font-weight: bold;
        }
      ")
    )
  ),

  # Application title
  titlePanel("电子申报验证标准检查"),

  # Sidebar with a slider input
  sidebarLayout(
    sidebarPanel(textAreaInput("file1", label=HTML("请输入待检查文件夹路径"),

value=r"(C:\Users\documents\p999mk9999\m5)"),
                 textAreaInput("file2", label=HTML("如发现问题，细节希望输出在以
下路径"),
                               value=r"(C:\Users\documents\submission-
check)"),
                 actionButton("go", "输出检查结果"),
                 actionButton("check", "检查电子签章")),

    # Show a plot of the generated distribution
    mainPanel(
      navbarPage(
      title = "",
      id = "nav",
      tabPanel("检查结果",
               div(textOutput("text1"), class = "my-text"),
               DTOutput("info"),
               div(textOutput("text2"), class = "my-text"),
               DTOutput("error")
      ),
      tabPanel("验证标准与模拟路径",
               div(textOutput("text3"), class = "my-text"),
               DTOutput("rule"),
```

```
                div(textOutput("text4"), class = "my-text"),
                DTOutput("path")
        ),
        tabPanel("电子签章",
                plotOutput("loopPlot", width = "100%", height = "800px")
        )
      ))

)))
```

SUPPORTIVE OUTPUTS

Due to limited space on the UI page of this R Shiny app, detailed information about which folders or files have violated the rules is output to an Excel file and saved in the output folder path as user specifies automatically.

As shown in Display 3, there is a sheet named "问题总结" (Problem Summary) that provides a summary of the number of errors encountered for each rule. If there is at least one file or folder that violates a rule, a new sheet will be created, and a hyperlink will be added to the description of the rule in the "问题总结" sheet. This hyperlink allows users to directly access the sheet listing all the files or folders that have violated that particular rule.

	A	B	C
1	No.	问题描述	数量
2	2.1	文件夹为空	1
3	2.2	文件夹与文件在同一级目录中	1
4	2.3	不能超出文件大小限制	0
5	2.4	文件类型（文件扩展名检查）	11
6	2.5	文件或文件夹名称不符合要求	3
7	2.6	文件夹和文件名（含扩展名）路径长度必须符合要求	3
8	4.1	PDF文件必须可读。	1
9	4.2	PDF文件不能有密码保护	1
10	4.3	PDF版本必须正确	0
11	4.4	不允许带附件的PDF文件	0
12	4.7	大于5页的文件建议添加书签	0
13	4.24	如使用非标准字体，需嵌入在PDF文件中	3
14			
15			
16			

问题总结 | 文件夹为空 | 文件夹与文件在同一级目录中 | 文件类型（文件扩展名检查） | 文件或文件夹名称不符合要求 | 文件夹和文件名

Display 3. "问题总结" Sheet from the additional output Excel file

CONCLUSION

This R Shiny application can provide a really helpful check to confirm whether the NMPA conformance rules are followed for an electronic disc submission package and avoid the errors that lead to the rejection of a submission. It could serve as a pre-check and proof of quality control by statistical programming function before we make the data package drop-off to regulatory affairs, while the NMPA official software might not be available.

More rules (mainly “Prompt” level) will be added to the application in future releases to provide a more universal checking.

REFERENCES

Center for Drug Evaluation, NMPA. “国家药监局药审中心关于更新《申报资料电子光盘技术要求》等文件的通知”. December 11, 2023. Available at https://www.cde.org.cn/main/news/viewInfoCommon/2969c293179bd697dbb64c454926dd80.

ACKNOWLEDGMENTS

The authors would like to thank Ting Li, Shushu Fang, and Weiwei Jiao from MSD China for their tremendous support, effective advice, and helpful input to this study from the regulatory affairs operation and programming perspectives.

We are also grateful for the support and review by leadership team of our department:

Amy Gillespie, AVP, head of MSD statistical programming, and

Greg Zhou, executive director, MSD statistical programming A&R Group 3.

CONTACT INFORMATION

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

Junfan Zhang
MSD China
jun.fan.zhang@merck.com