

# Interpreting Data Cutoff Methods Using R

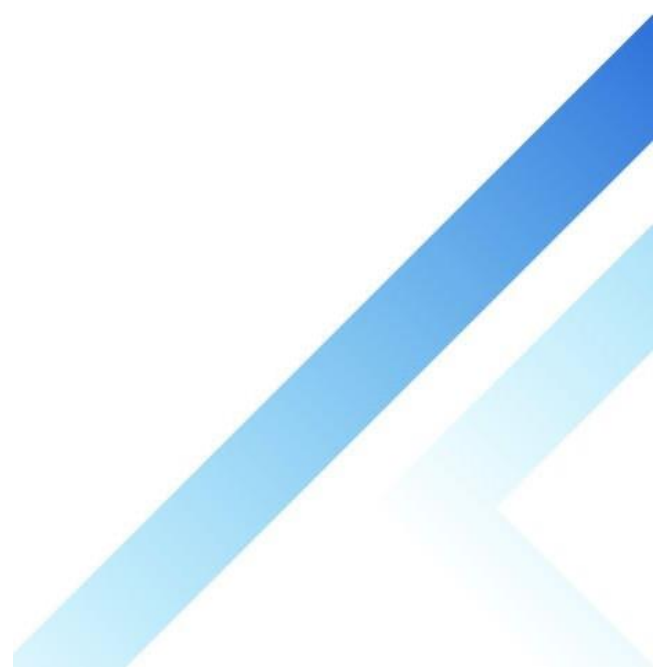
SM07 – PHUSE EU Connect 2024

---

Presenter: Lusine Barseghyan  
*Associate Manager, Statistical Programming*

# Agenda

---

- Introduction
  - DCO Process Using R
  - Step 1: Creating DCO Rules Metadata
  - Step 2: Execution in R
  - Practical Examples
  - Conclusion
  - Q&A
- 

# Introduction

---

- **Importance of DCO**
  - Essential for safeguarding participant safety in clinical trials
  
- **Challenges with Traditional Methods**
  - Cumbersome and time-consuming processes (e.g., SAS)
  - Increased risk of errors due to manual handling
  
- **Transition to R**
  - Flexible, efficient solutions that streamlines data management and analysis

## R vs. SAS: Simplicity and Open-Source Power

---

- **Simplicity and Flexibility in Code Management**

- Updating macros can be difficult; users must modify specific logic for new features.
- In R, you can add new functions without rewriting or adjusting the entire code.

- **Real-Time Debugging and Immediate Feedback**

- Debugging nested or complex macros in SAS can be time-consuming and error-prone.

- **R as Open-Source**

- R is free to use, modify, and share, encouraging collaboration and customization.

# DCO Process with R

---

## Proposed Method

- Implementation of an efficient DCO process through a dedicated R package.

## Key Features: `execute_dco`

- Applies predefined DCO methods to selected datasets and variables.
- Customizable templates for user-defined methods and variables.

## **Step 1: Creating DCO Rules Metadata**

---

- **Input Metadata File**
  - Serves as the foundation for DCO implementation
- **Indispensable Identification**
  - Every study must identify relevant datasets and variables for DCO
- **Format**
  - Utilizes Excel (.xlsx) for user-friendly data entry and management

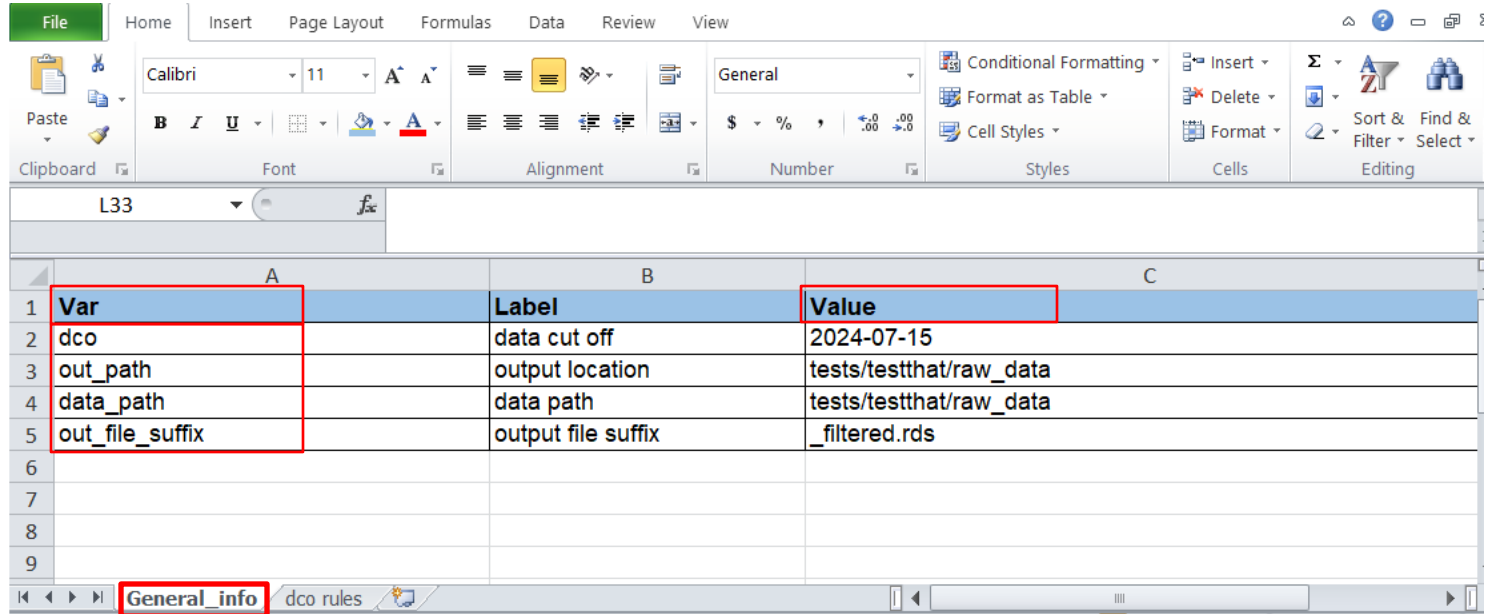
## General Information Tab

---

- |                             |                                                       |
|-----------------------------|-------------------------------------------------------|
| ▪ <b>DCO Date</b>           | Date designated for the Data Cutoff                   |
| ▪ <b>Data Path</b>          | File path to the raw data sources                     |
| ▪ <b>Output Location</b>    | Directory for saving results from the DCO process     |
| ▪ <b>Output File Suffix</b> | Suffix added to output files including file extension |

# Example: DCO Metadata

## General Information Tab: *General\_info*



| Var             | Label              | Value                   |
|-----------------|--------------------|-------------------------|
| dco             | data cut off       | 2024-07-15              |
| out_path        | output location    | tests/testthat/raw_data |
| data_path       | data path          | tests/testthat/raw_data |
| out_file_suffix | output file suffix | _filtered.rds           |
|                 |                    |                         |
|                 |                    |                         |
|                 |                    |                         |
|                 |                    |                         |

## DCO Rules Tab

---

**Dataset Names**

Raw dataset slated for DCO processing

**Variable Names**

Variables relevant to the cutoff implementation

**DCO Methods**

The name of an R function from the package

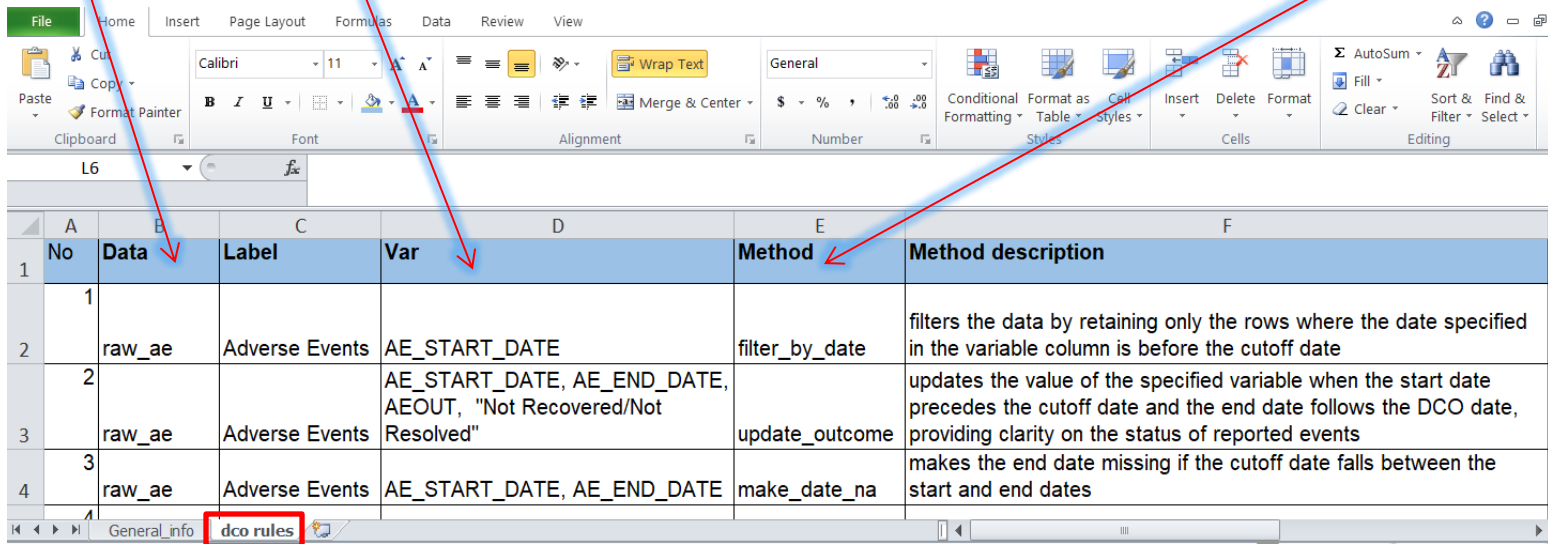
# Example: DCO Metadata

Variables & values

DCO Rules Tab: *dco\_rules*

Dataset Name

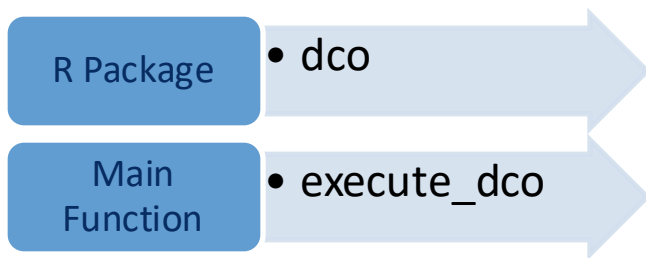
R function



| No | Data   | Label          | Var                                                             | Method         | Method description                                                                                                                                                                 |
|----|--------|----------------|-----------------------------------------------------------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | raw_ae | Adverse Events | AE_START_DATE                                                   | filter_by_date | filters the data by retaining only the rows where the date specified in the variable column is before the cutoff date                                                              |
| 2  | raw_ae | Adverse Events | AE_START_DATE, AE_END_DATE, AEOUT, "Not Recovered/Not Resolved" | update_outcome | updates the value of the specified variable when the start date precedes the cutoff date and the end date follows the DCO date, providing clarity on the status of reported events |
| 3  | raw_ae | Adverse Events | AE_START_DATE, AE_END_DATE                                      | make_date_na   | makes the end date missing if the cutoff date falls between the start and end dates                                                                                                |

## Step 2: Execution in R

---



### Parameter Overview

```
1 metadata_path <- test_path("metadata/DCO_rules.xlsx")
2 execute_dco(metadata_path)
```

## R Functions for DCO

---

### Overview of Key Functions

| Method         | Method description                                                                                        |
|----------------|-----------------------------------------------------------------------------------------------------------|
| filter_by_date | Retains rows where the specified date is before the cutoff date.                                          |
| update_outcome | Updates specified variable if the start date is before the cutoff and the end date is after the DCO date. |
| make_date_na   | Sets the end date as missing if the cutoff falls between the start and end dates.                         |

## Step 2: Execution in R

*filter\_by\_date*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY |
|---|------------|----------------|---------------|-------------|-------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | 2024-08-10  | Mild        |
| 2 | SUBJ001    | Anxiety        | 2024-07-04 ✖  | 2024-08-06  | Severe      |
| 3 | SUBJ002    | Headache       | 2024-06-20    | 2024-08-09  | Moderate    |
| 4 | SUBJ002    | Abdominal Pain | 2024-06-30 ✖  | 2024-08-09  | Moderate    |
| 5 | SUBJ002    | Dizziness      | 2024-07-08 ✖  | 2024-08-09  | Mild        |
| 6 | SUBJ003    | Insomnia       | 2024-07-06 ✖  | 2024-08-06  | Severe      |

*DCO date: 2024-06-25*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY |
|---|------------|----------------|---------------|-------------|-------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | 2024-08-10  | Mild        |
| 2 | SUBJ002    | Headache       | 2024-06-20    | 2024-08-09  | Moderate    |
| 3 | SUBJ005    | Abdominal Pain | 2024-06-02    | 2024-08-07  | Moderate    |
| 4 | SUBJ005    | Dizziness      | 2024-06-13    | 2024-08-06  | Mild        |

## Step 2: Execution in R

*update\_outcome*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY | AE_RELATIONSHIP  | AE_OUTCOME                                                                                        |
|---|------------|----------------|---------------|-------------|-------------|------------------|---------------------------------------------------------------------------------------------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | 2024-08-10  | Mild        | Possibly Related | Not Recovered  |
| 2 | SUBJ001    | Anxiety        | 2024-07-04    | 2024-08-06  | Severe      | Not Related      | Recovered                                                                                         |
| 3 | SUBJ002    | Headache       | 2024-06-20    | 2024-08-09  | Moderate    | Not Related      | Recovered      |
| 4 | SUBJ002    | Abdominal Pain | 2024-06-30    | 2024-08-09  | Moderate    | Related          | Not Recovered  |
| 5 | SUBJ002    | Dizziness      | 2024-07-08    | 2024-08-09  | Mild        | Possibly Related | Not Recovered  |
| 6 | SUBJ003    | Insomnia       | 2024-07-06    | 2024-08-06  | Severe      | Related          | Not Recovered                                                                                     |

*DCO date: 2024-08-07*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY | AE_RELATIONSHIP  | AE_OUTCOME                 | AE_SERIOUS |
|---|------------|----------------|---------------|-------------|-------------|------------------|----------------------------|------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | 2024-08-10  | Mild        | Possibly Related | Not Recovered/Not Resolved | No         |
| 2 | SUBJ001    | Anxiety        | 2024-07-04    | 2024-08-06  | Severe      | Not Related      | Recovered                  | No         |
| 3 | SUBJ002    | Headache       | 2024-06-20    | 2024-08-09  | Moderate    | Not Related      | Not Recovered/Not Resolved | No         |
| 4 | SUBJ002    | Abdominal Pain | 2024-06-30    | 2024-08-09  | Moderate    | Related          | Not Recovered/Not Resolved | No         |
| 5 | SUBJ002    | Dizziness      | 2024-07-08    | 2024-08-09  | Mild        | Possibly Related | Not Recovered/Not Resolved | No         |
| 6 | SUBJ003    | Insomnia       | 2024-07-06    | 2024-08-06  | Severe      | Related          | Not Recovered              | Yes        |

## Step 2: Execution in R

*make\_date\_na*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY | AE_RELATIONSHIP  | AE_OUTCOME    | AE_SERIOUS |
|---|------------|----------------|---------------|-------------|-------------|------------------|---------------|------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | 2024-08-10  | Mild        | Possibly Related | Not Recovered | No         |
| 2 | SUBJ001    | Anxiety        | 2024-07-04    | 2024-08-06  | Severe      | Not Related      | Recovered     | No         |
| 3 | SUBJ002    | Headache       | 2024-06-20    | 2024-08-09  | Moderate    | Not Related      | Recovered     | No         |
| 4 | SUBJ002    | Abdominal Pain | 2024-06-30    | 2024-08-09  | Moderate    | Related          | Not Recovered | No         |
| 5 | SUBJ002    | Dizziness      | 2024-07-08    | 2024-08-09  | Mild        | Possibly Related | Not Recovered | No         |
| 6 | SUBJ003    | Insomnia       | 2024-07-06    | 2024-08-06  | Severe      | Related          | Not Recovered | Yes        |

*DCO date: 2024-08-07*

|   | SUBJECT_ID | AE_TERM        | AE_START_DATE | AE_END_DATE | AE_SEVERITY | AE_RELATIONSHIP  | AE_OUTCOME    | AE_SERIOUS |
|---|------------|----------------|---------------|-------------|-------------|------------------|---------------|------------|
| 1 | SUBJ001    | Fatigue        | 2024-06-20    | NA          | Mild        | Possibly Related | Not Recovered | No         |
| 2 | SUBJ001    | Anxiety        | 2024-07-04    | 2024-08-06  | Severe      | Not Related      | Recovered     | No         |
| 3 | SUBJ002    | Headache       | 2024-06-20    | NA          | Moderate    | Not Related      | Recovered     | No         |
| 4 | SUBJ002    | Abdominal Pain | 2024-06-30    | NA          | Moderate    | Related          | Not Recovered | No         |
| 5 | SUBJ002    | Dizziness      | 2024-07-08    | NA          | Mild        | Possibly Related | Not Recovered | No         |
| 6 | SUBJ003    | Insomnia       | 2024-07-06    | 2024-08-06  | Severe      | Related          | Not Recovered | Yes        |

## Horizontal Data Structure

| Data      | Label          | Var                                          | Method         |
|-----------|----------------|----------------------------------------------|----------------|
| raw_ae    | Adverse Events | AE_START_DATE_1, AE_END_DATE_1, AETERM_1, NA | update_outcome |
| raw_ae    | Adverse Events | AE_START_DATE_2, AE_END_DATE_2, AETERM_2, NA | update_outcome |
| raw_ae    | Adverse Events | AE_START_DATE_3, AE_END_DATE_3, AETERM_3, NA | update_outcome |
| raw_ae    | Adverse Events | AE_START_DATE_4, AE_END_DATE_4, AETERM_4, NA | update_outcome |
| raw_ae    | Adverse Events | AE_START_DATE_5, AE_END_DATE_5, AETERM_5, NA | update_outcome |
| raw_ae    | Adverse Events | AE_START_DATE_6, AE_END_DATE_6, AETERM_6, NA | update_outcome |
| dc0 rules | General_info   |                                              |                |

## Date Formats

---

```
#' Convert Dates to YYYY-MM-DD Format
#
#' This function takes a date string in various formats (e.g., "DD-MM-YYYY", "
#' and converts it to a standardized "YYYY-MM-DD" format.
#
#' @param date_str A character string representing a date in formats such as "
#' @return A character string representing the date in "YYYY-MM-DD" format.
#
#' @importFrom lubridate parse_date_time
#' @export
convert_dates_to_ymd <- function(date_str) {
  formats <- c("dmy", "dmY", "dB Y", "mdy", "Ymd")
  parse_date_time(date_str, orders = formats) %>%
    format("%Y-%m-%d") %>% as.Date()
}
```

## Partial Dates

| Var             | Label                  | Value                   |
|-----------------|------------------------|-------------------------|
| dco             | data cut off           | 2024-07-15              |
| out_path        | output location        | tests/testthat/raw_data |
| data_path       | data path              | tests/testthat/raw_data |
| out_file_suffix | output file suffix     | filtered.rds            |
| impute_frmt     | imputation date format | "%Y-%m-%d", "%d%b%Y"    |
| impute_mtd      | imputation method      | end, start              |

ISO8601DA

DATE9

## How it Works

Main packages

Main function

```

1 #' Apply Methods from Excel File to Data Frames
2 #'
3 #' @description Reads an Excel file and using variables and their values applies
4 #' the specified methods from doc rules to data frames.
5 #'
6 #' @param dco_xlsx_path The path to the dco Excel file.
7 #'
8 #' @importFrom dplyr filter sym
9 #' @importFrom readxl read_excel
10 #'
11 #' @return Maintains the data frame derived from the applied functions(in RDS format).
12 #'
13 #' @export
14 execute_dco <- function(dco_xlsx_path) {
15
16   # Read the Excel file into a data frame
17   general_info_df <- read_excel(dco_xlsx_path, sheet = "General_info")
18
19   for (i in seq_len(nrow(general_info_df))) {
20     var_name <- general_info_df$Var[i]
21     var_value <- general_info_df$Value[i]
22     # Assign the value to a variable with the name `var_name`
23     assign(var_name, var_value)
24   }
25
26   apply_methods(dco_xlsx_path, data_path, out_path, out_file_suffix, dco)
27 }
28

```

## How it Works

---

```
13 #' @export
14 #'
15 apply_methods <- function(xlsx_path, data_path, out_path, out_file_suffix = "_filtered.rds", dco = "2024-07-13") {
16   if (nchar(dco) != 10) {
17     stop("Invalid date length for dco (yyyy-mm-dd format)")
18   }
19   # Read the Excel file into a data frame
20   method_df <- read_excel(xlsx_path, sheet = "dco_rules")
21   # Store data names from excel sheet without duplicates
22   data_names <- unique(method_df$Data)
23   # Iterate over each data frame from data_names and filter
24   for (j in (1:length(data_names))) {
25     method_df_loop <- method_df |> filter(method_df$Data == data_names[j])
26     # Construct the full path to the data file (assuming .rds format)
27     data_file_path <- file.path(data_path, paste0(method_df_loop$Data[j], ".rds"))
28     # Load the data frame from the specified path
29     data <- readRDS(data_file_path)
30     data <- data %>%
31       mutate(
```

# Conclusion

---

- **Efficient DCO Implementation**
  - Introduces a streamlined approach for Data Cutoff processes in clinical trials using R.
- **User-Friendly Metadata**
  - Enables easy organization of critical information about datasets and variables through a simplified metadata file.
- **Automation with `execute_dco`**
  - Automates the DCO process, reducing complexity compared to traditional manual methods like SAS.
- **Room for Improvement**
  - Potential for custom functions to enhance flexibility and adaptability across diverse research needs.

# Thank You Questions?



lusine.barseghyan@statecs.com



+374 11 666 101



<https://statecs.com>

