Certara R School

Lesson 2: R Basics for Pharmacometrics

May 25th, 2022

Keith Nieforth,
Senior Director, Software Product Management

James Craig,
R Software Engineer and Shiny Developer

Certara R School

Lesson 3: Exploratory Data Analysis (EDA) Using ggquickeda

6/29/2022 @10am EST

Register for Lesson 3

CERTARA

What will you learn?

Today we focus on data import and manipulation

- PK Data Overview
  - Creating an R project for your work and Importing files
  - RsNLME Dataset Structure
- Create a RsNLME dataset from CDISC SDTM
  - DM – Demographics
  - EX – Exposure
  - PC – PK Concentrations
  - VS – Vital Signs
- Useful R package `dplyr` functions for data wrangling
- Descriptive statistics
  - Mean concentration by time
  - Demography summary
- Exploratory graphics with ggplot2
  - Concentration-time by subject
  - Mean concentration-time by dose group

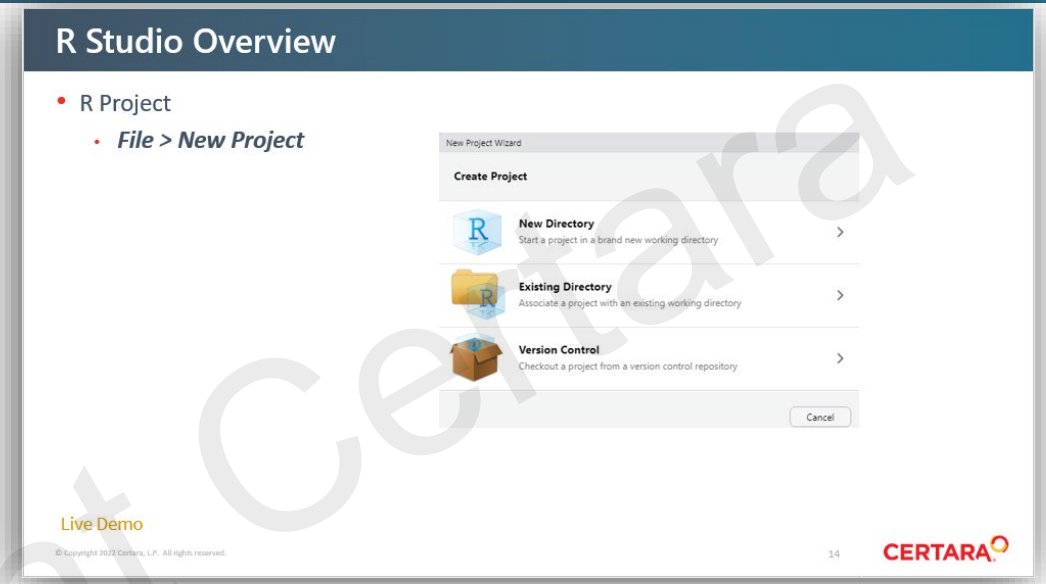
We will be assembling a script and demonstrating some best practices for this work

CERTARA

Recap From Last Lecture

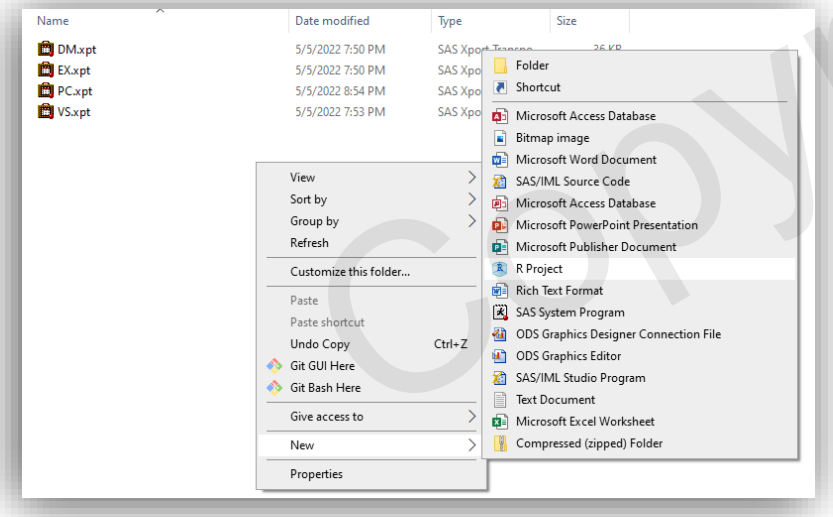
Creating an R project for your work

- Last week we covered how to create a new project with RStudio GUI



pRo Tip!

- Create a project with right click in Windows



https://www.anthonyschmidt.co/post/2021-11-19-creating-a-new-r-project-from-the-windows-context-menu/

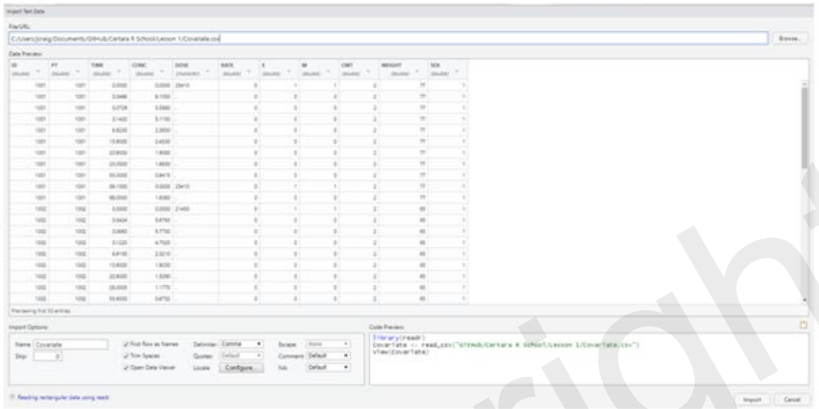
CERTARA

Recap from Last Lecture

Importing Files

R Studio Overview

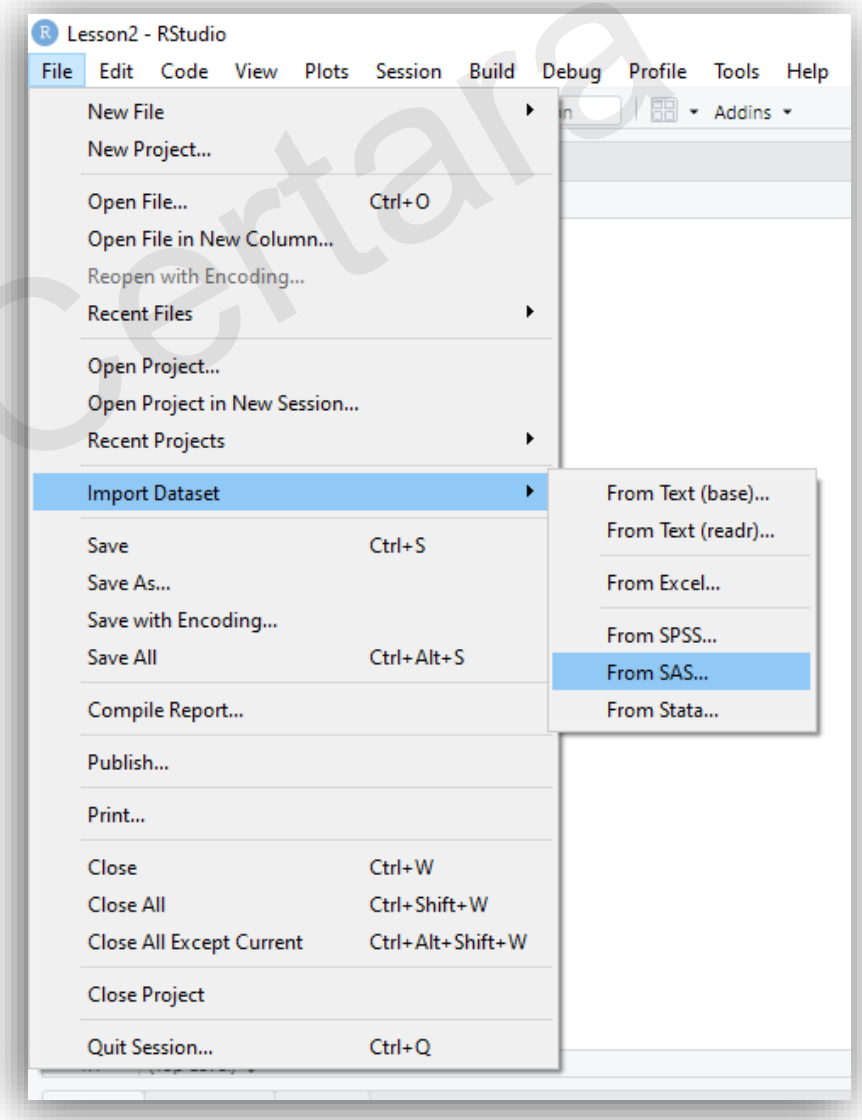
- Importing Data
  - ***File > Import Dataset***
  - Import xls
    - Define NA
  - Import CSV
    - Define col types
    - Copy code, update NA argument
  - Importing data with whitespace delimiter



Live Demo

16

CERTARA



CERTARA

STDM Source Files

DM: Demography Information

STUDYID	DOMAIN	USUBJID	SUBJID	RFSTDTC	RFENDTC	RFXSTDTC	RFXENDTC	RFICDTC	RFPENDTC	DTHDTC	DTHFL	SITEID	AGE	AGEU	SEX	RACE	ETHNIC	ARMCD	ARM
SMPSTDYPK	DM	SMPSTDYPK-000001	1	2021-03-02		2021-03-02	2021-06-13	2021-03-01	2021-07-04			123456	56	YEARS	M	WHITE	HISPANIC OR LATINO	A	Treatment Regimen A
SMPSTDYPK	DM	SMPSTDYPK-000002	2	2021-03-02		2021-03-02	2021-06-13	2021-03-01	2021-07-04			123456	45		M	WHITE	NOT HISPANIC OR LATINO	A	Treatment Regimen A
SMPSTDYPK	DM	SMPSTDYPK-000003	3	2021-03-02		2021-03-02	2021-06-13	2021-03-01	2021-07-04			123456	51		M	NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER	NOT HISPANIC OR LATINO	A	Treatment Regimen A
SMPSTDYPK	DM	SMPSTDYPK-000004	4	2021-03-02		2021-03-02	2021-06-13	2021-03-01	2021-07-04			123456	47		F	WHITE	NOT HISPANIC OR LATINO	A	Treatment Regimen A

EX: Dosing Information

STUDYID	DOMAIN	USUBJID	EXSEQ	EXTRT	EXDOSE	EXDOSU	EXDOSFRM	EXDOSFRQ	EXROUTE	VISITNUM	VISIT	EPOCH	EXSTDTC	EXENDTC	EXSTDY	EXENDY
SMPSTDYPK	EX	SMPSTDYPK-000001	1	Drug A	5000	mg	TABLET	SINGLE DOSE	ORAL	20	WEEK 1	TREATMENT	2021-03-02T08:00	2021-03-02T08:00	1	1
SMPSTDYPK	EX	SMPSTDYPK-000002	1	Drug A	5000	mg	TABLET	SINGLE DOSE	ORAL	20	WEEK 1	TREATMENT	2021-03-02T08:00	2021-03-02T08:00	1	1
SMPSTDYPK	EX	SMPSTDYPK-000003	1	Drug A	5000	mg	TABLET	SINGLE DOSE	ORAL	20	WEEK 1	TREATMENT	2021-03-02T08:00	2021-03-02T08:00	1	1
SMPSTDYPK	EX	SMPSTDYPK-000004	1	Drug A	5000	mg	TABLET	SINGLE DOSE	ORAL	20	WEEK 1	TREATMENT	2021-03-02T08:00	2021-03-02T08:00	1	1
SMPSTDYPK	EX	SMPSTDYPK-000005	1	Drug A	5000	mg	TABLET	SINGLE DOSE	ORAL	20	WEEK 1	TREATMENT	2021-03-02T08:00	2021-03-02T08:00	1	1

PC: PK (PD) Sampling Information

STUDYID	DOMAIN	USUBJID	PCSEQ	PCREFID	PCTESTCD	PCTEST	PCORRES	PCORRESU	PCSTRESC	PCSTRESN	PCSTRESU	PCSTAT	PCREASND	PCNAM	PCSPEC	PCLLOQ	VISITNUM	VISIT
SMPSTDYPK	PC	SMPSTDYPK-00...	1	S001345670	AN001	ANALYTE DRU...	BLQ	ug/mL	BLQ		ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	2	S001345671	AN001	ANALYTE DRU...	8.61	ug/mL	8.61	8.61	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	3	S001345672	AN001	ANALYTE DRU...	19.4	ug/mL	19.4	19.4	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	4	S001345673	AN001	ANALYTE DRU...	34	ug/mL	34	34	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	5	S001345674	AN001	ANALYTE DRU...	30.2	ug/mL	30.2	30.2	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	6	S001345675	AN001	ANALYTE DRU...	31.3	ug/mL	31.3	31.3	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	7	S001345676	AN001	ANALYTE DRU...	25	ug/mL	25	25	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1
SMPSTDYPK	PC	SMPSTDYPK-00...	8	S001345677	AN001	ANALYTE DRU...	23.4	ug/mL	23.4	23.4	ug/mL			BLUE BELL LABS	PLASMA	0.1	20	WEEK 1

VS: Vital Sign Information

STUDYID	DOMAIN	USUBJID	VSSEQ	VSTESTCD	VSTEST	VSPOS	VSORRES	VSORRESU	VSSTRESC	VSSTRESN	VSSTRESU	VSSTAT	VSREASND	VSBLFL	VSDRVFL	VISITNUM	VISIT	EPOCH	VSDTC	VSDY
SMPSTDYPK	VS	SMPSTDYPK-000001	1	WEIGHT	Weight		95	kg	95	95	kg			Y		10	SCREENING	SCREENING	2021-03-01	-1
SMPSTDYPK	VS	SMPSTDYPK-000001	2	WEIGHT	Weight		94	kg	94	94	kg					20	WEEK 1	TREATMENT	2021-03-02	1
SMPSTDYPK	VS	SMPSTDYPK-000002	1	WEIGHT	Weight		65	kg	65	65	kg			Y		10	SCREENING	SCREENING	2021-03-01	-1
SMPSTDYPK	VS	SMPSTDYPK-000002	2	WEIGHT	Weight		64	kg	64	64	kg					20	WEEK 1	TREATMENT	2021-03-02	1
SMPSTDYPK	VS	SMPSTDYPK-000003	1	WEIGHT	Weight		69	kg	69	69	kg			Y		10	SCREENING	SCREENING	2021-03-01	-1
SMPSTDYPK	VS	SMPSTDYPK-000003	2	WEIGHT	Weight		68	kg	68	68	kg					20	WEEK 1	TREATMENT	2021-03-02	1
SMPSTDYPK	VS	SMPSTDYPK-000004	1	WEIGHT	Weight		60	kg	60	60	kg			Y		10	SCREENING	SCREENING	2021-03-01	-1
SMPSTDYPK	VS	SMPSTDYPK-000004	2	WEIGHT	Weight		62	kg	62	62	kg					20	WEEK 1	TREATMENT	2021-03-02	1
SMPSTDYPK	VS	SMPSTDYPK-000005	1	WEIGHT	Weight		72	kg	72	72	kg			Y		10	SCREENING	SCREENING	2021-03-01	-1

CERTARA

Demo – R Script Section 1: Import Data

```
# Section 1: Import Data ----
```

CERTARA

A Note on R Data Objects

Some fundamentals to be aware of

- A **vector** is the most basic data object in R
  - There are six “types”, but for now we look at three
  - A vector can only contain one data type
    - Concept know as “atomic” vector

double

```
> x<-c(1.1, 3.5, 2.7)
> x
[1] 1.1 3.5 2.7
> typeof(x)
[1] "double"
```

character

```
> x<-c("tree","dog")
> x
[1] "tree" "dog"
> typeof(x)
[1] "character"
```

logical

```
> x<-c(T,F)
> x
[1]  TRUE FALSE
> typeof(x)
[1] "logical"
```

A mixed vector will “coerce” to character

```
> x<-c(1.1, 3.5, 2.7, "dog", "cat", T, F)
> x
[1] "1.1"   "3.5"   "2.7"   "dog"   "cat"   "TRUE"  "FALSE"
> typeof(x)
[1] "character"
```

- A **matrix** is collection of equal length vectors arranged in a two-dimensional rectangular layout
  - A matrix can only contain one data type

matrix

```
> x<-c(1.1, 3.5, 2.7)
> y<-c(4, 5.6, 6)
> z<-cbind(x,y)
> z
        x   y
[1,] 1.1 4.0
[2,] 3.5 5.6
[3,] 2.7 6.0
> typeof(z)
[1] "double"
> class(z)
[1] "matrix" "array"
```

only one data type allowed

```
> x<-c(1.1, 3.5, 2.7)
> y<-c("tree", "dog", "cat")
> z<-cbind(x,y)
> z
     x     y
[1,] "1.1" "tree"
[2,] "3.5" "dog"
[3,] "2.7" "cat"
> typeof(z)
[1] "character"
> class(z)
[1] "matrix" "array"
```


CERTARA

A Note on R Data Objects (cont.)

Some fundamentals to be aware of

- A **data frame** is collection of equal length vectors arranged in a two-dimensional rectangular layout.
  - (a matrix) but….
  - A data frame can contain multiple data types
  - Most commonly used data object for PMX type data

data frame

```
> x<-c(1.1, 3.5, 2.7)
> y<-c("tree", "dog", "cat")
> z<-data.frame(x,y)
> z
    x    y
1 1.1 tree
2 3.5  dog
3 2.7  cat
> typeof(z)
[1] "list"
> class(z)
[1] "data.frame"
```

A “list” is an R object that contains elements of different types

- A **tibble** is a data frame
  - Tweaked to optimize performance in the tidyverse
  - Fewer “default” actions
    - Never converts strings to factors or changes the names of variables
  - More flexibility
    - Can use non-compliant column names (numbers, spaces in names)
    - …but need to reference these “non-syntactic” names with backticks ``
  - Printing
    - Will only display first 10 rows of large data objects
    - Displays column type
    - Color and font style enhanced display

tibble

```
> x<-c(1.1, 3.5, 2.7)
> y<-c("tree", "dog", "cat")
> z<-tibble(x,y)
> z
# A tibble: 3 x 2
      x y
  <dbl> <chr>
1   1.1 tree
2   3.5 dog
3   2.7 cat
> typeof(z)
[1] "list"
> class(z)
[1] "tbl_df"     "tbl"        "data.frame"
```


CERTARA

Let’s Create a RsNLME Dataset!

Study Description

Basic Design and Source Data

- 150 Subjects
- 3 Treatment Groups
  - Single 5, 10, or 20 mg oral dose administered at time=0
- Serial blood samples for PK collected up to 24 hours post-dose
- Desired covariate information in dataset:
  - Age, Weight, Gender, Race, Dose Group

- Source data are SDTM “Standard Data Tabulation Model” datasets

ID	Time	Amt	Conc	Age	Weight	Gender	Race	DoseGrp
1	0	5000	0	56	94	Male	WHITE	5000
1	0.25	0	8.612809	56	94	Male	WHITE	5000
1	0.5	0	19.43682	56	94	Male	WHITE	5000
1	1	0	34.0067	56	94	Male	WHITE	5000
1	2	0	30.2288	56	94	Male	WHITE	5000
1	3	0	31.29961	56	94	Male	WHITE	5000
1	4	0	24.97912	56	94	Male	WHITE	5000
1	6	0	23.37618	56	94	Male	WHITE	5000
1	8	0	23.51309	56	94	Male	WHITE	5000
1	12	0	14.67865	56	94	Male	WHITE	5000
1	16	0	9.073463	56	94	Male	WHITE	5000
1	24	0	5.2962	56	94	Male	WHITE	5000
2	0	5000	0	45	64	Male	WHITE	5000
2	0.25	0	30.40228	45	64	Male	WHITE	5000
2	0.5	0	47.62615	45	64	Male	WHITE	5000
2	1	0	64.42828	45	64	Male	WHITE	5000
2	2	0	100.1783	45	64	Male	WHITE	5000
2	3	0	92.56154	45	64	Male	WHITE	5000
2	4	0	65.26794	45	64	Male	WHITE	5000
2	6	0	49.26316	45	64	Male	WHITE	5000
2	8	0	60.28592	45	64	Male	WHITE	5000
2	12	0	44.24852	45	64	Male	WHITE	5000
2	16	0	41.85031	45	64	Male	WHITE	5000
2	24	0	32.89788	45	64	Male	WHITE	5000

CERTARA

RsNLME Dataset Format Overview

Event by Event History of Dosing, Observations, and Covariate Values

- Each row of the dataset is an “event”
  - Dosing Events
  - Observation Events
  - Other events (covariate value change, etc.)
- Required Columns
  - PK Model
    - A1 or Aa (amount administered central or extravascular)
    - Time
    - CObs (observed concentration)
  - Emax or Linear Model
    - C (independent variable)
    - Eobs (observed effect)
- Many optional columns
  - Steady state, additional, dose interval, rate, duration, mdv, blq, etc.
  - Advanced options covered in a future lecture

ID	Time	Amt	Conc	Age	Weight	Gender	Race	DoseGrp
1	0	5000	0	56	94	Male	WHITE	5000
1	0.25	0	8.612809	56	94	Male	WHITE	5000
1	0.5	0	19.43682	56	94	Male	WHITE	5000
1	1	0	34.0067	56	94	Male	WHITE	5000
1	2	0	30.2288	56	94	Male	WHITE	5000
1	3	0	31.29961	56	94	Male	WHITE	5000
1	4	0	24.97912	56	94	Male	WHITE	5000
1	6	0	23.37618	56	94	Male	WHITE	5000
1	8	0	23.51309	56	94	Male	WHITE	5000
1	12	0	14.67865	56	94	Male	WHITE	5000
1	16	0	9.073463	56	94	Male	WHITE	5000
1	24	0	5.2962	56	94	Male	WHITE	5000
2	0	5000	0	45	64	Male	WHITE	5000
2	0.25	0	30.40228	45	64	Male	WHITE	5000
2	0.5	0	47.62615	45	64	Male	WHITE	5000
2	1	0	64.42828	45	64	Male	WHITE	5000
2	2	0	100.1783	45	64	Male	WHITE	5000
2	3	0	92.56154	45	64	Male	WHITE	5000
2	4	0	65.26794	45	64	Male	WHITE	5000
2	6	0	49.26316	45	64	Male	WHITE	5000
2	8	0	60.28592	45	64	Male	WHITE	5000
2	12	0	44.24852	45	64	Male	WHITE	5000
2	16	0	41.85031	45	64	Male	WHITE	5000
2	24	0	32.89788	45	64	Male	WHITE	5000

CERTARA

RsNLME is flexible on requirements for input data

- Can use any column names you like (except reserved)
- Can use date-times as time variables
  - No need to derive numeric times
- Can use characters for variable values (e.g., M/F)
- Up to 5 ID variables
  - Analysis across multiple studies with repeating ID's?
    - No problem, use Study and ID as ID variables
- Dose information need not be integrated into same dataset as observations
- Covariate values can be missing in dataset
  - Can "fill in" missing covariate data
    - Carry forward
    - Carry backwards
    - Interpolate
- RsNLME does not require time to be sorted
  - Will sort automatically within ID

Caveats

- "Good data housekeeping" is critical to quality analyses
  - Flexible data requirements can allow you to work in a messy house
  - Errors harder to see, unexpected results

- Suggested practice for new modelers
  - Use derived actual times in your dataset
    - 1st event (usually a dose) is time=0, all subsequent events relative to this
  - Avoid long strings for character variables if present
    - No spaces, abbreviate

      Can always enhance labeling for displays
  - Mind your units!
    - e.g., dose in ug, conc in ug/mL, time in hr will give parameter units in mL and hr

      CL=mL/hr, V=hr

CERTARA

Common Data Manipulation Commands in dplyr

```
> df
  ID Treatment Dose
1  1       TRT A    5
2  2     Placebo   NA
3  3       TRT B   10
4  4       TRT C   20
5  5     Placebo   NA
```

arrange:

```
> df %>% arrange(Dose)
  ID Treatment Dose
1  1       TRT A    5
2  3       TRT B   10
3  4       TRT C   20
4  2     Placebo   NA
5  5     Placebo   NA

> df %>% arrange(desc(Dose))
  ID Treatment Dose
1  4       TRT C   20
2  3       TRT B   10
3  1       TRT A    5
4  2     Placebo   NA
5  5     Placebo   NA
```

rename:

```
> df %>% rename("AMT" = Dose)
  ID Treatment AMT
1  1       TRT A   5
2  2     Placebo  NA
3  3       TRT B  10
4  4       TRT C  20
5  5     Placebo  NA
```

select:

```
> df %>% select(ID,Dose)
  ID Dose
1  1    5
2  2   NA
3  3   10
4  4   20
5  5   NA
```

filter:

```
> df %>% filter(Treatment != "Placebo")
  ID Treatment Dose
1  1       TRT A    5
2  3       TRT B   10
3  4       TRT C   20
```

mutate:

```
> df %>% mutate(DoseMCG = Dose*1000)
  ID Treatment Dose DoseMCG
1  1       TRT A    5    5000
2  2     Placebo   NA      NA
3  3       TRT B   10   10000
4  4       TRT C   20   20000
5  5     Placebo   NA      NA
```

if_else:

```
> df %>% mutate(DoseMCG = if_else(Treatment!="Placebo",Dose*1000, 0))
  ID Treatment Dose DoseMCG
1  1       TRT A    5    5000
2  2     Placebo   NA       0
3  3       TRT B   10   10000
4  4       TRT C   20   20000
5  5     Placebo   NA       0
```

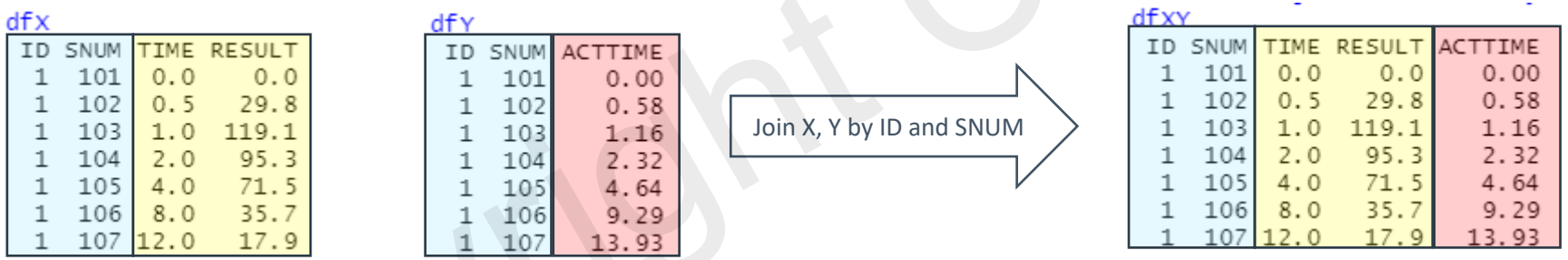

CERTARA

Combining Data From Different Objects

To merge dataframes, use a “join”

- What is a “join”?
- Have 2 dataframes, x and y
- A join command says I want to add (merge) columns from dataframe y to dataframe x, matching on some key

- Example:



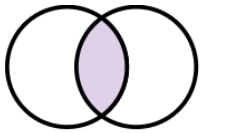
Joins add columns from y to x, matching rows based on the keys:

- inner_join(): includes all rows in x and y
- left_join(): includes all rows in x
- right_join(): includes all rows in y
- full_join(): includes all rows in x or y

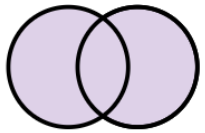
CERTARA

Joins add columns from Y to X, matching rows based on keys

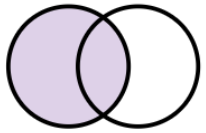
General `dplyr` merging syntax: join(X, Y, by=key(s))



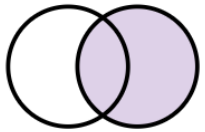
- inner_join
  - Only keep rows where key exists in both X and Y



- full_join
  - Keeps all rows from X and Y



- left_join
  - Keeps all rows from X
  - Most commonly used join



- right_join
  - Keeps all rows from Y

X	Y	Add from Y to X

```
> df1
  ID Treatment
1  1       TRT A
2  2     Placebo
3  3       TRT B
4  4       TRT C
5  5     Placebo
```

```
df2
  ID Dose
1  1    5
2  3   10
3  4   20
4  6   40
```

```
inner_join(df1,df2,by="ID")
  ID Treatment Dose
1  1       TRT A    5
2  3       TRT B   10
3  4       TRT C   20
```

```
> df1
  ID Treatment
1  1       TRT A
2  2     Placebo
3  3       TRT B
4  4       TRT C
5  5     Placebo
```

```
df2
  ID Dose
1  1    5
2  3   10
3  4   20
4  6   40
```

```
full_join(df1,df2,by="ID")
  ID Treatment Dose
1  1       TRT A    5
2  2     Placebo   NA
3  3       TRT B   10
4  4       TRT C   20
5  5     Placebo   NA
6 NA        <NA>   40
```

```
> df1
  ID Treatment
1  1       TRT A
2  2     Placebo
3  3       TRT B
4  4       TRT C
5  5     Placebo
```

```
df2
  ID Dose
1  1    5
2  3   10
3  4   20
4  6   40
```

```
left_join(df1,df2,by="ID")
  ID Treatment Dose
1  1       TRT A    5
2  2     Placebo   NA
3  3       TRT B   10
4  4       TRT C   20
5  5     Placebo   NA
```

```
> df1
  ID Treatment
1  1       TRT A
2  2     Placebo
3  3       TRT B
4  4       TRT C
5  5     Placebo
```

```
df2
  ID Dose
1  1    5
2  3   10
3  4   20
4  6   40
```

```
right_join(df1,df2,by="ID")
  ID Treatment Dose
1  1       TRT A    5
2  3       TRT B   10
3  4       TRT C   20
4  6        <NA>   40
```


CERTARA

Reshaping

Pivot wider and pivot longer

- pivot_longer makes datasets longer by increasing the number of rows and decreasing the number of columns.

```
> df
# A tibble: 3 x 8
     ID   `0` `0.5`   `1`   `2`   `4`   `8`  `12`
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1     1     0  27.1 108.   86.7  65    32.5  16.3
2     2     0  21.6  86.5  69.2  51.9  25.9  13
3     3     0  22.2  88.9  71.1  53.3  26.7  13.3
> df %>% pivot_longer(!ID, names_to = "TIME", values_to = "RESULT")
# A tibble: 21 x 3
     ID TIME  RESULT
  <dbl> <chr>  <dbl>
1     1 0       0
2     1 0.5    27.1
3     1 1     108.
4     1 2      86.7
5     1 4      65
6     1 8      32.5
7     1 12     16.3
8     2 0       0
9     2 0.5    21.6
10    2 1      86.5
# ... with 11 more rows
```

- pivot_wider makes a dataset wider by increasing the number of columns and decreasing the number of rows

```
> dfx
   ID TIME RESULT
1   1  0.0    0.0
2   1  0.5   27.1
3   1  1.0  108.3
4   1  2.0   86.7
5   1  4.0   65.0
6   1  8.0   32.5
7   1 12.0   16.3
8   2  0.0    0.0
9   2  0.5   21.6
10  2  1.0   86.5
11  2  2.0   69.2
12  2  4.0   51.9
13  2  8.0   25.9
14  2 12.0   13.0
15  3  0.0    0.0
16  3  0.5   22.2
17  3  1.0   88.9
18  3  2.0   71.1
19  3  4.0   53.3
20  3  8.0   26.7
21  3 12.0   13.3
> dfx %>% pivot_wider(id_cols = ID, names_from = TIME, values_from = RESULT)
# A tibble: 3 x 8
     ID   `0` `0.5`   `1`   `2`   `4`   `8`  `12`
  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1     1     0  27.1 108.   86.7  65    32.5  16.3
2     2     0  21.6  86.5  69.2  51.9  25.9  13
3     3     0  22.2  88.9  71.1  53.3  26.7  13.3
```


CERTARA

Demo – R Script Section 2: Dataset Preparation

```
# Section 2: Dataset Preparation ----
```

CERTARA

Tables in R : gtsummary or flextable?

- `gtsummary` tables offer most comprehensive functionality, combining data summary operations with table output
  - Lacks output support across all core types
  - Only support for one grouping variable
- `flextable` does not perform data summary operations, simply converts as is `data.frame` to table output
  - `flextable` does support all core output types however
- More complex data summary operations should be handled with `dplyr`, then convert the resulting `data.frame` to a `flextable` object
- `gtsummary` table objects can also be converted to `flextable`, allowing you to export the table to different output types

Print Engine	Function	HTML	PDF	RTF	Word
gt	as_gt()	😁	🚧	🚧	⛔
flextable	as_flex_table()	😁	😁	⛔	😁
huxtable	as_hux_table()	😁	😁	😁	😁
kableExtra	as_kable_extra()	😁	😁	⛔	⛔
kable	as_kable()	😐	😐	😐	😐
tibble	as_tibble()	☹	☹	☹	☹

Key	
😁	Output fully supported
😐	Formatted output, but missing indentation, footnotes, spanning headers
☹	No formatted output
⛔	Output not supported
🚧	Under development

CERTARA

Demo – R Script Section 3: Descriptive Statistics Tables

```
# Section 3: Descriptive Statistics Tables ----
```

CERTARA

Exploratory Data Analysis with `ggplot2`

- One of the core strengths of R is support for data visualization
- Many plotting packages to choose from:
  - Base R e.g., `plot()`, `hist()`
  - `lattice`
  - `ggplot2`
  - Pharmacometric visualization packages
    - `xpose4`, `xpose`, `Certara.Xpose.NLME`, `ggcertara`
- `ggplot2` offers the most comprehensive plotting functionality in R
- Start with ggplot(), supply dataset and aesthetic mappings with aes(), then add layers, and optionally split plots (facet) by one or more columns with discrete values.
  - aes
    - x
    - y
    - *color*
    - *group*
  - layers
    - geom_point()
    - geom_line()
    - geom_bar

</>

```
ggplot(data, aes(x,y)) +
   geom_point()
```


CERTARA

Demo – R Script Section 4: Exploratory Data Analysis

```
# Section 4: Exploratory Data Analysis ----
```

CERTARA

Questions

CERTARA

Certara R School

Lesson 3: Exploratory Data Analysis (EDA) Using ggquickeda

5/25/2022 @10am EST

[Register for Lesson 3]

CERTARA