Paper SD09

Demonstrations of AutoTable

Chengxin Li, AutoCheng Clinical Data Services LLC

ABSTRACT

This paper presents AutoTable, an automated analysis-generation toolset implemented entirely in SAS macros, where parameters are aligned with Analysis Results Metadata (ARM) fields. The AutoTable system presented here focuses on efficiently and reproducibly generating non-inferential analysis tables, integrating automation directly into existing statistical-programming workflows.

Four core macros have been developed—%ADSLEVL, %BDSSTAT, %BDSSHIFT, and %OCCFREQ—each supporting a different category of non-inferential analysis. For every macro invocation, the following outputs are automatically generated:

- Table (RTF, PDF, or XLSX formats)
- Plain, executable, submission-ready SAS code
- Final result dataset (RDS)
- Associated metadata (ARMDS), facilitating define-xml generation
- Log file

During the live demonstration, four representative call instances will be showcased:

1. Demographic table using %ADSLEVL.
2. SOC/PT/Severity table using %OCCFREQ
3. Observed values and CHG/PCHG by Visit table using %BDSSTAT.
4. Lab shift table using %BDSSHIFT.

Leveraging ARM for analyses, the tools work with any table-shell layout by capturing its core structural features and layout patterns, thus Table Shell Standardization or Digitalization

NOT a Prerequisite. With autoTable, programming efficiency is greatly enhanced—significantly more efficient than the manual copy-paste-update workflows commonly used in legacy programming, possibly switched from hours to minutes for a table.

INTRODUCTION

Automating table, listing, and figure (TLF) generation have long been a major objective in statistical programming. Traditional workflows depend heavily on manual coding, copying, and incremental updates of legacy programs. This process is time-consuming and error-prone, particularly when managing frequent updates across multiple tables or studies.

To address these limitations, the AutoTable toolkit was developed—a metadata-driven automation framework that uses Analysis Results Metadata (ARM) to configure reusable SAS macros. Each macro call becomes a transparent, metadata-aligned instruction capable of producing submission-ready outputs.

The framework aims to automate repetitive analysis tables with submission-ready outputs, including plain executable SAS code. To support the multiplicity of table layouts, key features and layout patterns are configurable through macro options.

ANALYSIS RESULTS METADATA (ARM)

Overview of ARM

Analysis Results Metadata (ARM) is defined in ADaM v2.1 (December 2009**)** as one of three core metadata types:

- ✓ Analysis Dataset Metadata
- ✓ Analysis Variable Metadata
- ✓ **Analysis Results Metadata**

ARM documents the key attributes of each analysis result and provides traceability between analysis outputs and their ADaM datasets to facilitate reviews. AutoTable toolsets extend the ARM as the framework to non-inferential analyses.

AutoTLF Driven by ARM

Achieving autoTLF requires ADaM or pre-processing ADaM to be analysis-ready. After applying data-selection criteria (WHERE statement), the system can directly invoke SAS statistical procedures, formatting, and reporting. Figure 1 illustrates the link between programming process and ARM.

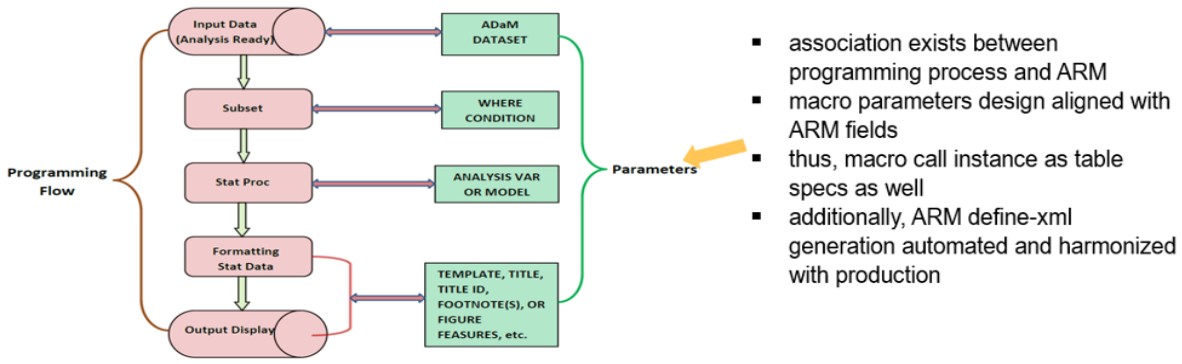

Figure 1. Statistical Programming process Aligned with ARM

ARM drives AutoTable's automation across multiple dimensions:

- **Parameterization:** ARM fields directly configure SAS macros
- **Traceability:** ARM provides direct linkage between analysis results, ADaM datasets, and outputs, ensuring full transparency for regulatory submission.
- **Layout control:** ARM attributes capture layout multiplicity and patterns of table shells, eliminating the need for rigid table-shell standardization or digitalization.

By integrating ARM into both analysis and documentation, AutoTable ensures that metadata consistency and traceability are maintained throughout the study lifecycle.

AUTOTABLE FRAMEWORK

Design Overview

AutoTable is implemented entirely in SAS macros, designed to generate analysis outputs automatically based on ARM metadata. The framework presented here focuses on non-inferential analyses, targeting 95% of analyses and 85% of table layouts in clinical studies.

Each macro call uses parameter values aligned with ARM fields, ensuring full consistency between metadata, specifications, and outputs. AutoTable supports flexible configuration of both table-level and parameter-level options—such as page breaks, sort orders, and formatting patterns.

Macro Implementation

Four core macros have been developed covering non-inferential analyses as described below table 1:

Table 1. AutoTable Scopes for Non-inferential Analysis

SAS Macro	Input ADaM	Analyses
%OCCFREQ	Any OCCDS, e.g., ADAE, ADCM, ADMH	Any frequency analyses
%BDSSTAT	Any BDS, e.g., ADLB, ADEG, ADVS	Any descriptive analyses with continuous/character variables
%BDSSHIFT	Any BDS, e.g., ADLB, ADEG	Shift table analyses, specifically
%ADSLEVL	ADSL	Subject evaluation analyses, e.g., tables of baseline, demographic, disposition, population, etc.

Table 2 outlines the outputs from each macro call automatically.

Table 2. AutoTable Output

Seq	Item	Note
1	**Table**	table output in formats of RTF, PDF, or XLSX
2	**SAS Codes**	Plain executable submission-ready SAS code dynamically resolved from the macro call. Executing the generated code reproduces the same table output as produced by the macro call
3	**RDS**	Final results dataset used for qc, or implementing any other special table layout
4	**ARMDS**	Aggregated dataset of table specifications with ARM structure Facilitate ARM define-xml generations
5	**Log file**	Macro execution logs

Transparency and Flexibility

Unlike black-box automation tools, AutoTable macros are fully transparent. Each macro call directly reflects the table specification, with parameters representing table definitions. The macros are compatible with any table-shell layout by capturing core structural features—header patterns, grouping variables, and layout patterns.

Because all outputs (including metadata and code) are generated simultaneously, programmers can review, validate, and trace every component of the workflow.

DEMONSTRATIONS

With simulated ADaM datasets (ADSL, ADLB, and ADAE), four live demonstrations illustrate AutoTable's practical implementation. For any macro call instance, the corresponding plain

executable and submission-ready SAS codes in a txt file is resolved. The same table will be produced if execute the codes (omitted here).

1. **Demographic Table (%ADSLEVL)**

```
%ADSLEVL(
       TABLEID=%NRBQUOTE(t_demo2\LCHLBL=Category), /*TableID required; LCHLBL option to update the default display label*/
       TITLE=,    /*opt., separated with # if multiple. if both TITLE and FOOTNOTE NULL indicating user with utility macro*/
       FOOTNOTE=, /*opt., separated with # if multiple*/

       POPU=SAFFL, /*POPU parameter for big N summary as well*/

       ADSL=%str(ADaM.ADSL),
       WHERE =%str(SAFFL='Y'),   /*data selection criteria from ADSL, applied to all analysis variables lised in ANALVAR parameters*/

       ADSL_ARM=%NRBQUOTE(TRT01A\PRELOADFMT=$TRTA.\INFORMAT=TRTAN.\THCLIST=Xanomeline|[TRT11 TRT12 TRT19] ~ TRT21 ~ TRT99\THCFMT=$TRTSHEADER.),
                          /*PRELOADFMT option for trt grouping, THCLIST option for spanning header, THCFMT option for further fomatting*/

       ANALVAR=%NRBQUOTE(AGEGR1\CATSUM=OLI\DISP=%nrbquote(Age (years), n (%)) # /*CATSUM option for variable layout pattern*/
                         AGE\DISP=Age (years) # /*DISP option for variable label display, by default, with variable label*/
                         ARACE\CATSUM=OLI\preloadfmt=$RACE.\informat=RACEN.\DISP=%nrbquote(Race, n (%)) #
                         AETHNIC\CATSUM=OLI\DISP=%nrbquote(Ethnicity, n (%)) #
                         WEIGHTBL # HEIGHTBL # BMIBL # ACOUNTRY\DISP=Country\informat=COUNTRYN. /*informat option for sort*/
                         )  /*CHAR var triggering freq summary, NUM var triggering DSTAT summary*/
   );
```

By default, the output table is in RTF format. Below is the partial screen shot of produced demo table.

Category	XANOMELINE			Placebo (N=86)	All Subjects (N=254)
	Low Dose (N=84)	High Dose (N=84)	Pooled (N=168)		
Age (years), n (%)					
<65	8 (9.5)	11 (13.1)	19 (11.3)	14 (16.3)	33 (13.0)
65-80	47 (56.0)	55 (65.5)	102 (60.7)	42 (48.8)	144 (56.7)
>80	29 (34.5)	18 (21.4)	47 (28.0)	30 (34.9)	77 (30.3)
Age (years)					
n	84	84	168	86	254
Mean (SD)	75.7 (8.29)	74.4 (7.89)	75.0 (8.09)	75.2 (8.59)	75.1 (8.25)
Median	77.5	76.0	77.0	76.0	77.0
Min, Max	51, 88	56, 88	51, 88	52, 89	51, 89

Several options are provided for the user to adjust the number of data lines per page, the width of left (Category) column, the display label of left column, and page break patterns.

For Population Flag related analysis variables, some conventions are further defined. For instance, if POPU variable (denominator) also listed in ANALVAR, any variable prior to the POPU is only summarized with ‘n’; the variable prior to RANDFL is summarized with ‘n’ only at treatment TOTAL group.

2. SOC/PT/Severity Table (%OCCFREQ)

```
%OCCFREQ(
    TABLEID=%NRBQUOTE(t_teae_socptsev), /*left column header autogenerated by default, further customerized if needed*/
    Title  =%str(Table 14.3.1.2 Treatment-Emergent Adverse Events by System Organ Class, Preferred Term and Severity # (Safety Analysis Set)),
    Footnote=%NRBQUOTE(TEAE: treatment-emergent adverse event. #
                       Note: Adverse events (AEs) are coded using MedDRA version 27.0.),
                      /*commonly titles/footnotes loaded with company utility macro instead of manual input here*/
    Popu   =%str(SAFFL),
    Indata =ADaM.ADAE,     /*or WORK.ADAE if preprocessed*/
    ADSL   =ADaM.ADSL,
    SUBJSUM=,              /*%str(EVENT\DISP=No. of Events # USUBJ\DISP=Subjects with at least one TEAE)*/

    ANALVAR=%NRBQUOTE(ASEV\WHERE=AOCCIFL='Y'\preloadfmt=$ASEV.\informat=ASEVN.\CATSUM=OLI\disp=TEAE #
                      AESOC\DISP=System Organ Class\INDENT=0\SUMOFF=Y #   /*INDENT option defining embeded layer associations*/
                        ASEV\WHERE=AOCCSIFL='Y'\preloadfmt=$ASEV\informat=ASEVN.\INDENT=1 #
                        AEDECOD\DISP=Preferred Term\INDENT=1\SUMOFF=Y #
                          ASEV\WHERE=AOCCPIFL='Y'\preloadfmt=$ASEV.\informat=ASEVN.\INDENT=2
                      ),

    ADSL_ARM=%str(TRT01A),
    INDATA_ARM=%NRBQUOTE(TRTA\PRELOADFMT=$TRTA.\INFORMAT=TRTAN.\THCLIST="Xanomeline"|[TRT11 TRT12 TRT19] ~ TRT21 ~ TRT99\THCFMT=$TRTSHEADER.)
);  /*THCLIST for table spanning header, TCHFMT further formating header display if needed*/
```

Below is the partial screen shot of the produced table.

Table 14.3.1.2 Treatment-Emergent Adverse Events by System Organ Class, Preferred Term and Severity
(Safety Analysis Set)

System Organ Class Preferred Term Analysis Severity/Intensity	Xanomeline Low Dose (N=84)	Xanomeline High Dose (N=84)	Xanomeline Pooled (N=168)	Placebo (N=86)	All Subjects (N=254)
TEAE					
Mild	0	2 (2.4)	2 (1.2)	3 (3.5)	5 (2.0)
Moderate	41 (48.8)	37 (44.0)	78 (46.4)	41 (47.7)	119 (46.9)
Severe	36 (42.9)	40 (47.6)	76 (45.2)	25 (29.1)	101 (39.8)
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS					
Mild	1 (1.2)	0	1 (0.6)	1 (1.2)	2 (0.8)
Moderate	39 (46.4)	37 (44.0)	76 (45.2)	19 (22.1)	95 (37.4)
Severe	7 (8.3)	3 (3.6)	10 (6.0)	1 (1.2)	11 (4.3)
APPLICATION SITE PRURITUS					
Mild	0	0	0	0	0
Moderate	18 (21.4)	20 (23.8)	38 (22.6)	6 (7.0)	44 (17.3)
Severe	4 (4.8)	2 (2.4)	6 (3.6)	0	6 (2.4)

Requested by the table shell, with the display pattern option setting, the severity criteria (ASEV) can be transposed to header display under treatment groups, like below**:**

System Organ Class Preferred Term	Xanomeline Low Dose (N=84) Mild n %	Xanomeline Low Dose (N=84) Moderate n %	Xanomeline Low Dose (N=84) Severe n %	Xanomeline High Dose (N=84) Mild n %	Xanomeline High Dose (N=84) Moderate n %	Xanomeline High Dose (N=84) Severe n %	Placebo (N=86) Mild n %	Placebo (N=86) Moderate n %	Placebo (N=86) Severe n %
TEAE	0	41 (48.8)	36 (42.9)	2 (2.4)	37 (44.0)	40 (47.6)	3 (3.5)	41 (47.7)	25 (29.1)
GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS	1 (1.2)	39 (46.4)	7 (8.3)	0	37 (44.0)	3 (3.6)	1 (1.2)	19 (22.1)	1 (1.2)
APPLICATION SITE PRURITUS	0	18 (21.4)	4 (4.8)	0	20 (23.8)	2 (2.4)	0	6 (7.0)	0

The SOC/PT Severity table is the most complicated table with OCCDS. Any other frequency table with OCCDS can be easily generated.

3. **Observed and Change from Baseline Table (%BDSSTAT)**

```
%BDSSTAT(
    TABLEID=%str(t_lab_saf),

    POPU=SAFFL,  /*SAFFL is default population as well*/

    INDATA=ADaM.ADLB,
    where=%str(AVISITN>=0),
    ADSL=ADaM.ADSL,

    ANALVAR=%NRBQUOTE(BASE\disp=Baseline # AVAL # CHG),
    BYVAR=%str(PARAM\DPAT=TV # AVISIT\DPAT=LC), /*DPAT option making support more flexible table shells*/

    ADSL_ARM=TRT01A,
    INDATA_ARM=%NRBQUOTE(TRTA\PRELOADFMT=$TRTA.\INFORMAT=TRTAN.)
    );
```

Below is the partial screen shot of the produced table.

Parameter = Hemoglobin (mmol/L)

Analysis Visit Statistics	Xanomeline Low Dose (N=84)	Xanomeline High Dose (N=84)	Pooled Xanomeline (N=168)	Placebo (N=86)	All Subjects (N=254)
Baseline					
n	81	81	162	85	247
Mean (SD)	8.62 (0.768)	8.87 (0.783)	8.75 (0.784)	8.62 (0.828)	8.70 (0.800)
Median	8.56	8.87	8.75	8.63	8.69
Min, Max	6.6, 10.2	7.4, 10.6	6.6, 10.6	6.4, 10.4	6.4, 10.6
Week 2					
n	80	80	160	83	243
Mean (SD)	8.45 (0.763)	8.72 (0.829)	8.58 (0.805)	8.44 (0.765)	8.53 (0.793)
Median	8.44	8.66	8.53	8.50	8.50
Min, Max	6.5, 10.5	7.1, 10.9	6.5, 10.9	6.8, 10.4	6.5, 10.9
Change from Baseline					
n	77	77	154	82	236
Mean (SD)	-0.16 (0.462)	-0.21 (0.425)	-0.18 (0.443)	-0.16 (0.416)	-0.17 (0.433)
Median	-0.06	-0.25	-0.19	-0.19	-0.19
Min, Max	-1.2, 0.7	-1.3, 1.2	-1.3, 1.2	-1.1, 0.9	-1.3, 1.2

The table shell may request the analysis variables (AVAL, CHG, or PCHG) transposed as the table header, and the display label at left column also possibly different with the default label. Correspondingly, the macro call instances:

```
%BDSSTAT(
    TABLEID=%NRBQUOTE(t_lab_saf2\LCHLBL=%NRBQUOTE(Visit^n   Statistics)\AVARDPAT=SH\LCPCHG=1%),
    /*analysis variables transposed to header display with option setting, left column width/label adjustable*/
    POPU=SAFFL,

    INDATA=ADaM.ADLB,
    where=%str(AVISITN>=0),
    ADSL=ADaM.ADSL,

    ANALVAR=%NRBQUOTE(BASE\disp=Baseline # AVAL\DISP=Observed Value # CHG),
    /*multi BASE/AVAL/CHG/PCHG for study PART1 and PART2 easily achieved e.g., with WHERE suboption: BASETYPE='...'*/


    BYVAR=%str(PARAM\DPAT=TV # AVISIT),     /*DPAT option supporting more flexible table shells*/

    ADSL_ARM=TRT01A,
    INDATA_ARM=%NRBQUOTE(TRTA\PRELOADFMT=$TRTA.\INFORMAT=TRTAN.)
);
```

Below is the partial screen shot of the produced table:

Parameter = Hemoglobin (mmol/L)

Visit Statistics	Xanomeline Low Dose (N=84)		Xanomeline High Dose (N=84)		Pooled Xanomeline (N=168)		Placebo (N=86)		All Subjects (N=254)	
	Observed Value	Change from Baseline	Observed Value	Change from Baseline	Observed Value	Change from Baseline	Observed Value	Change from Baseline	Observed Value	Change from Baseline
Baseline										
n	81		81		162		85		247	
Mean (SD)	8.62 (0.768)		8.87 (0.783)		8.75 (0.784)		8.62 (0.828)		8.70 (0.800)	
Median	8.56		8.87		8.75		8.63		8.69	
Min, Max	6.6, 10.2		7.4, 10.6		6.6, 10.6		6.4, 10.4		6.4, 10.6	
Week 2										
n	80	77	80	77	160	154	83	82	243	236
Mean (SD)	8.45 (0.763)	-0.16 (0.462)	8.72 (0.829)	-0.21 (0.425)	8.58 (0.805)	-0.18 (0.443)	8.44 (0.765)	-0.16 (0.416)	8.53 (0.793)	-0.17 (0.433)
Median	8.44	-0.06	8.66	-0.25	8.53	-0.19	8.50	-0.19	8.50	-0.19
Min, Max	6.5, 10.5	-1.2, 0.7	7.1, 10.9	-1.3, 1.2	6.5, 10.9	-1.3, 1.2	6.8, 10.4	-1.1, 0.9	6.5, 10.9	-1.3, 1.2
Week 4										
n	71	69	71	69	142	138	79	78	221	216
Mean (SD)	8.35 (0.780)	-0.30 (0.501)	8.62 (0.780)	-0.21 (0.452)	8.49 (0.789)	-0.25 (0.477)	8.37 (0.815)	-0.23 (0.436)	8.45 (0.798)	-0.25 (0.462)
Median	8.19	-0.25	8.56	-0.25	8.38	-0.25	8.32	-0.31	8.38	-0.25
Min, Max	6.1, 10.1	-2.4, 0.9	7.4, 10.8	-1.2, 1.0	6.1, 10.8	-2.4, 1.0	6.5, 10.1	-1.1, 1.1	6.1, 10.8	-2.4, 1.1

The number of statistic decimals can be auto-determined from the observed values by PARAM, e.g., decimals in min/max kept the same as the observed values, mean/medium with +1, SD/SE with +2. Optionally, it can be denoted by the user as well.

Multi page-break patterns are defined for quick turnaround.

4. **Laboratory Shift Table (%BDSSHIFT)**

The shift tables often show big flexibility with layouts. With settings, the SHIFT table engine can easily implement:

1) Either BASELINE or POST-BASELINE transposed
2) Denominator with N, m, or n
3) Combinations of TRTA, subgroup, PARAM, AVISIT presented in title (with byVALn), left column, or spanning header
4) Multiple shifts appended in one table

Conforming with table shell, below call instance generates one table with treatment group and VISIT displayed at left column, PARAM at title.

```
**********BDSSHIFT************************************************************;
proc format;
/*multilabel format widely used along with the borrowed SAS native PRELOADFMT option
 to implement overlaping, aggregated, or zero count display, and often paired with SAS informat*/
    value $NRIND(multilabel)
        'Low'='Low'
        'Normal'='Normal'
        'High'='High'
        'Low', 'Normal', 'High'='Total'
    ;

    invalue NRINDN
        'Low'=1
        'Normal'=2
        'High'=3
        'Total'=9
    ;
run;
```

```
/*with table shell requested, treatment group and visits displayed at left column; PARAM at title*/
%BDSSHIFT(
    TABLEID=t_lab_shift1,
    POPU=SAFFL,

    INDATA=%str(ADaM.ADLB),
    WHERE =%NRBQUOTE(PARAMCD in ('ALT', 'AST', 'CK')),

    ADSL=%str(ADaM.ADSL),

    ANALVAR=%NRBQUOTE(BNRIND\preloadfmt=$NRIND.\informat=NRINDN. ~ ANRIND\preloadfmt=$NRIND.\informat=NRINDN.),
    /*the 2nd paired variable(here ANRIND) always transposed. if BNRIDN transposed, then BNRIND listed to 2nd;
      multi-pairs supported with '#' separated, and further with WHERE sub-option enabled to select the data.*/

    BYVAR=%str(PARAMCD\DPAT=TV # AVISIT\DPAT=LC),

    ADSL_ARM=%str(TRT01A),
    INDATA_ARM=%str(TRTA\DPAT=LC)
);
```

The corresponding generated table with partial screen shot:

Parameter: Alanine Aminotransferase (U/L)

		Post-Baseline Category, n (%)			
Treatment Group	Baseline Category	Low	Normal	High	Total
Xanomeline Low Dose (N=84)	Week 2				
	Low	2 (2.4)	1 (1.2)	0	3 (3.6)
	Normal	0	73 (86.9)	0	73 (86.9)
	High	0	1 (1.2)	1 (1.2)	2 (2.4)
	Total	2 (2.4)	75 (89.3)	1 (1.2)	78 (92.9)
	Week 4				
	Low	2 (2.4)	0	0	2 (2.4)
	Normal	2 (2.4)	64 (76.2)	0	66 (78.6)
	High	0	1 (1.2)	1 (1.2)	2 (2.4)
	Total	4 (4.8)	65 (77.4)	1 (1.2)	70 (83.3)

On table shell request, by simply denoting option DENOM=m for post-baseline variable ANRIND, the denominator switched to overall counting by PARAM AVISIT. The macro call instance is shown below:

```
/*category overall baseline count--m as denominator, and postbase transposed*/
%BDSSHIFT(
    TABLEID=t_lab_shift3,
    POPU=SAFFL,

    INDATA=%str(ADaM.ADLB),
    WHERE =%NRBQUOTE(PARAMCD in ('ALT', 'AST', 'CK')),

    ADSL=%str(ADaM.ADSL),

    ANALVAR=%NRBQUOTE(BNRIND\preloadfmt=$NRIND.\informat=NRINDN. ~ ANRIND\preloadfmt=$NRIND.\informat=NRINDN.\denom=m),

    BYVAR=%str(PARAMCD\DPAT=TV # AVISIT\DPAT=LC),

    ADSL_ARM=%str(TRT01A),
    INDATA_ARM=%str(TRTA\DPAT=LC)
);
```

The corresponding part of generated table:

Parameter: Alanine Aminotransferase (U/L)

		Post-Baseline Category, n (%)			
Treatment Group	Baseline Category	Low	Normal	High	Total
Xanomeline Low Dose (N=84)	Week 2				
	Low	2 (2.6)	1 (1.3)	0	3 (3.8)
	Normal	0	73 (93.6)	0	73 (93.6)
	High	0	1 (1.3)	1 (1.3)	2 (2.6)
	Total	2 (2.6)	75 (96.2)	1 (1.3)	78 (100)
	Week 4				
	Low	2 (2.9)	0	0	2 (2.9)
	Normal	2 (2.9)	64 (91.4)	0	66 (94.3)
	High	0	1 (1.4)	1 (1.4)	2 (2.9)
	Total	4 (5.7)	65 (92.9)	1 (1.4)	70 (100)

AutoTLF ROADMAP

The AutoTable project is part of a broader **autoTLF** roadmap as shown below table 3.

Table 3. AutoTLF Roadmap

Stage	Scope	Status
Non-Inferential Tables	%ADSLEVL, %OCCFREQ, %BDSSTAT, %BDSSHIFT	Completed
Inferential Tables	%ANCOVA, %MMRM, %CMH, %TTE	In Progress
AI/ML Powered Agents	AI-based interpretation and automation	Initiated
Figures and Listings	Expansion beyond tables	Planned
R/Python Packages	Cross-language support	Under Evaluation

Future releases will extend automation to inferential analyses and integrate AI Agents for intelligent document interpretation. Long-term plans include migration toward multi-language support (SAS, R, Python) and hybrid architectures combining AI-driven reasoning with macro-based agents.

CONCLUSION

AutoTable demonstrates how metadata-driven automation can be implemented effectively within existing Biopharma statistical programming frameworks. By aligning SAS macro parameters with Analysis Results Metadata (ARM), the system achieves reliable, reproducible, and submission-ready table generation without requiring table-shell standardization or digitalization. However, it can facilitate the generations if standardization or digitalization already exists.

The approach offers transparency, efficiency, and regulatory traceability while maintaining full compatibility with legacy workflows. The designs summarize the programming practices by following programming styles, easily transitioned by seeing and knowing.

With the submission-ready executable plain SAS codes auto-resolved from the macro call, AutoTable substantially reduces manual programming effort—beating legacy copy-and-paste and program-update workflows—and lays the foundation for scalable, intelligent automation in clinical reporting.

CONTACT INFORMATION

Your comments and questions are valued and encouraged.

For demonstration and collaboration request, please contact the author at:

Author Name: Chengxin Li

Company: AutoCheng Clinical Data Services LLC

Email: Chengxin.li@autoclindata.com

All rights reserved.

Brand and product names are trademarks of their respective companies.