

Pharma SUG 2026 - Paper AP-295
**User-defined Functions for Programming Population PK and PKPD
Datasets**

Will He, Man Li, and Hong Yan, Regeneron Pharmaceuticals, Inc.

ABSTRACT

The Population PK (ADPPK) and Pharmacodynamic (ADPPKPD) datasets are the analysis-ready CDISC compliant datasets for population PK and Exposure-Response modeling analysis. These datasets are longitudinal, consist of various data elements with complexity across the clinical database domains, and involve many derivations in covariates (baseline and/or time-varying) in addition to dosing, PK, and PD, which often require using different calculations for a given covariate, or different covariates to facilitate different analyses during model development and selection process.

A set of user-defined functions was developed to reduce cycle time for dataset development. We will share several key functions which have been proven to be easy to use, accurate, and efficient in pharmacometrics data programming and output generating.

INTRODUCTION

SAS programmers frequently use macros or code snippets to perform common derivations and save time and avoid rewriting common patterns in the data step. By contrast, user-defined functions offer capabilities and flexibility such as can be invoked directly inside procedures (e.g.: PROC SQL, PROC FREQ etc.) and data step expressions, acting as native functions for computational logic and data manipulation.

User-defined functions are well suited for high-precision calculations and reusable logic that can be stored, unit-tested, and shared across programs as standard components. In population PK and exposure–response data processing and manipulation, often involves pooling of multiple studies across the projects or indication which requires performing unit conversions, executing complex calculations, and selecting right formular to align study level derivations for specific covariates. The validated user-defined functions can be used to reduce human error and improve programming efficiency.

USER-DEFINED FUNCTION OVERVIEW

The user-defined function is defined in PROC FCMP, compiled and stored in a dataset. This dataset needs to be included during data processing and computation. The details process is described below.

Step 1 Define a function:

```
PROC FCMP OUTLIB=work.funcs.xxx ;  
  FUNCTION xxx(a, b) ;  
    RETURN (a*b) ;  
  ENDSUB ;  
RUN ;
```

- PROC FCMP is the procedure used to compile and store user-defined functions.

- OUTLIB specifies the directory where the dataset containing functions is to be saved.
- FUNCTION xxx declares the function name xxx with its argument (a, b);
- RETURN specifies the value of function.
- ENDSUB or END FUNC marks the end of the function definition.

Step 2 Call function:

```
OPTIONS CMPLIB = work.funcs ;
```

- CMPLIB specifies datasets containing defined functions.

Step 3 Use user defined function:

```
DATA test2 ;
  SET test1 ;
  X = xxx(a, b) ;
RUN;
```

The function return value can be numeric or character string. Please note the value of use defined function is numeric by default. Users need to define the return value as character by adding \$ next to function.

```
PROC FCMP OUTLIB=work.funcs.xxx ;
  FUNCTION xxx(a, b) $ ;
  RETURN (a*b);
ENDSUB;
RUN;
```

It is worth mentioning that: dataset containing user defined functions need to be submitted to health authority along with the programs using the functions in submission package.

USER-DEFINED FUNCTION EXAMPLES

1. Function Calculating Renal Function (EGFR)

eGFR (estimated glomerular filtration rate) is a key indicator of kidney function, used as key baseline covariate in population PK analysis. eGFR is calculated based on age, sex, race, and serum creatine, three equations are commonly used.

- CKD-EPI 2009 creatinine equation: historically standard and includes race coefficient.

$$eGFR = 141 * \min\left(\frac{S_{cr}}{K}, 1\right)^\alpha * \max\left(\frac{S_{cr}}{K}, 1\right)^{-1.209} * 0.993^{Age} * (1.018 \text{ if female}) * (1.159 \text{ if Black})$$

Where Scr (serum creatinine)=mg/dL, K=0.7 (female) or 0.9 (Male), a = -0.329 (female) or -0.411 (male).

- CKD-EPI 2021 creatinine equation: current standard by removing the race coefficient.

$$eGFR = 142 * \min(S_{cr}/K, 1)^\alpha * \max(S_{cr}/K, 1)^{-1.200} * 0.9938^{Age} * (1.012 \text{ if female}),$$

where Scr (serum creatinine)=mg/dL, K=0.7 (female) or 0.9 (Male), a = -0.241 (female) or -0.302 (male).

- MDRD equation performs well in patients with impaired kidney function (GFR < 60 mL/min/1.73 m²) but is less accurate at higher GFR values.

$$eGFR = 175 * (Scr)^{-1.154} * (Age)^{-0.203} * (0.742 \text{ if female}) * (1.212 \text{ if Black})$$

To build the 3 different equations in one function, we add parameters to facilitate user picking and use:

```
proc fcmp outlib=funct.functions.egfr;
  function egfr(age,sexn,racen, scr, ckd09='N', ckd21='N', mdrd='N');
    if racen = 2 then afrc = 1 ;
    else afrc = 0 ;
    if mdrd = 'Y' then do ;
      gfr=175*(scr**(-1.154))*(age**(-0.203))*((sexn=1)+0.742*(sexn=2)) ;
      if afrc = 1 then gfr = gfr*1.212*(afrc=1);
    end;
    else if ckd09='Y' then do ;
      k = 0.7*(sexn=2) + 0.9*(sexn=1) ;
      a = (-0.329)*(sexn=2) + (-0.411) * (sexn=1) ;
      gfr = 141 * min(Scr/k, 1)**a*max(Scr/k, 1)**(-1.209*0.993**age ;
      if sexn=2 then gfr = gfr*1.018 ;
      if racen = 2 then gfr= gfr* 1.159;
    end;
    else if ckd21='Y' then do ;
      k= (sexn = 2)*0.7 + (sexn = 1)*0.9;
      a = (sexn =2)*(-0.241) + (sexn = 1)*(-0.302);
      gfr=142 * min(Scr/k, 1)**a * max(Scr/k, 1)**(-1.200) * 0.9938**age;
      if sexn=2 then gfr= gfr*1.012 ;
    end;

    if nmiss(age, scr, sexn, racen) = 0 then do ;
      return(gfr);
    end;
    else return(.) ;
  endsub;
run;
```

Note, this function requires sexn = 1 is male, sexn = 2 is female, racen = 2 is African American, Scr is serum creatine in mg/dL.

We can specify which equation you want to use when you use this function. For example,

```
data test2 ;
  set test ;
  ** using CKD-EPI 2021 equation;
  egfr_21 = egfr(age, sexn, racen, scr=creatbl, ckd21='Y' ;
  ***using MDRD equation ;
  egfr_mdrd = egfr(age, sexn, racen,scr=creatbl, mdrd='Y' ;
run;
```

2. Function Handling Significant Figures (SIGFIGS)

The significant figures are usually applied for summary statistics of drug concentration, target concentration, and PK parameters. There are different ways to handle this, such as Picture Format and Macros. These approaches require the update when the ranges are significant than predefined. We found the user defined function can easily cover the full range.

```
proc fcmp outlib=funct.functions.sigfigs;
  function sigfigs (n, invar) $32;
    length val $32;
    if invar then do;
      m=floor(log10(abs(invar)));
      rndvar=round(invar,10**(m-n+1));
      m=floor(log10(abs(rndvar)));
      val=if m>n-2 then left(put(rndvar,32.))
        else left(putn(rndvar,cat('32.',n-1-m)));
    end;
    else val=put(invar,1.);
    return(val);
  endsub;
run;
```

3. Function Handling Lean Body Weight and Mass (FFM, LBM_HU, LBW, LBW_JAM and LBW_PE)

Lean body mass (LBM), fat free mass (FFM), and lean body weight (LBW) are closely related concepts used to represent the nonfat portion of the body.

FFM: Boer LBM formula is often treated as FFM. It normalizes body fluid volumes for pharmacokinetic studies (Boer 1984).

$$\text{Men: } \text{FFM} = (0.407 * \text{Weight in kg}) + (0.267 * \text{Height in cm}) - 19.2$$

$$\text{Women: } \text{FFM} = (0.252 * \text{Weight in kg}) + (0.473 * \text{Height in cm}) - 48.3$$

```
proc fcmp OUTLIB=funct.functions.ffm;
  function ffm(kg, cm, sexn) ;
    if nmiss(kg,cm, sexn) = 0 then do;
      return ((sexn=1)*(0.407*kg+0.267*cm-19.2)+(sexn=2)*(0.252*kg+0.473*cm-48.3)) ;
    end;
    else return(.) ;
  endsub;
run;
```

Note, this function requires sexn = 1 is male, sexn = 2 is female, kg is body weight in Kg, cm is body height in cm.

LBM_HU : Hume formula, estimate LBM for creatine-clearance calculation (Hume 1966).

Men: $LBM = (0.32810 * \text{Weight in kg}) + (0.33929 * \text{Height in cm}) - 29.5336$

Women: $LBM = (0.29569 * \text{Weight in kg}) + (0.41813 * \text{Height in cm}) - 43.2833$

```
proc fcmp OUTLIB=funct.functions.lbm_hu;
  function lbm_hu(kg, cm, sexn) ;
    if nmiss(kg,cm, sexn) = 0 then do;
      return ((sexn=1)*(0.3281*kg+0.33929*cm-29.5336)+(sexn=2)*(0.29569*kg+0.41813*cm-
43.2833)) ;
    end;
    else return(.) ;
  endsub;
run;
```

Note, this function requires sexn = 1 is male, sexn = 2 is female, kg is body weight in Kg, cm is body height in cm.

LBW : Janmashtami (often spelled as Janmahasatian) formula, proxy for FFM, commonly used for medical /anesthesia dosing.

Men: $LBW = (9270 \times \text{Weight in kg}) / (6680 + 216 * \text{BMI})$

Women: $LBW = (9270 \times \text{Weight in kg}) / (8780 + 244 * \text{BMI})$

```
proc fcmp OUTLIB=funct.functions.lbw;
  function lbw(kg, bmi, sexn) ;
    if nmiss(kg,bmi, sexn) = 0 then do;
      return ((sexn=1)*((kg*9270)/(6680+216*BMI))+(sexn=2)*((kg*9270)/(8780+244*BMI))) ;
    end;
    else return(.) ;
  endsub;
run;
```

Note, this function requires sexn = 1 is male, sexn = 2 is female, kg is body weight in Kg, BMI is kg/m²

LBW_JAM : James's equation (Commonly Used), body-size descriptor in obesity research (James 1977).

Men: $1.10 * \text{Weight in kg} - 128 * \left(\frac{\text{Weight in kg}}{\text{Height in cm}}\right)^2$

Women: $1.07 * \text{Weight in kg} - 148 * \left(\frac{\text{Weight in kg}}{\text{Height in cm}}\right)^2$

```

proc fcmp outlib=funct.functions.lbw_jam ;
  function lbw_jam(kg, cm, sexn) ;
    if nmiss(kg,cm, sexn) = 0 then do;
      return ((sexn=1)*(1.1*kg-128*(kg/cm)**2)+(sexn=2)*(1.07*kg-148*((kg/cm)**2))) ;
    end;
    else return(.) ;
  endsub;
run;

```

Note, this function requires sexn = 1 is male, sexn = 2 is female, kg is body weight in Kg, cm is body height in cm.

LBW_PE: Peters (2011) – Children <= 14 years , sex-independent pediatric drug dosing (Peters 2011).

$$LBW = 3.8 * (0.0215 * \text{weight (kg)}^{0.6469} * \text{height (cm)}^{0.7236})$$

```

proc fcmp outlib=funct.functions.lbw_pe ;
  function lbw_pe(kg, cm) ;
    if nmiss(kg,cm) = 0 then do;
      return (3.8*(0.0215 *(kg)**0.6469 * ( cm)**0.7236)) ;
    end;
    else return(.) ;
  endsub;
run;

```

Note, this function requires kg is body weight in Kg, cm is body height in cm.

4. Function Calculating Hepatic Function (HEPTC)

Hepatic function is categorized by comparing total bilirubin and AST (aspartate aminotransferase) values with the laboratory-specific upper limits of normal (ULN) in the National Cancer Institute Organ Dysfunction Working Group (NCI-ODWG) hepatic function classification described below.

- Normal : total bilirubin ≤ULN and AST ≤ULN
- Mild : total bilirubin > ULN but ≤ 1.5 × ULN, or AST > ULN
- Moderate: total bilirubin > 1.5x-3xULN, regardless of AST
- Severe : total bilirubin > 3xULN, regardless of AST

```

proc fcmp OUTLIB=funct.functions.heptc ;
  FUNCTION heptc(bilibl, astbl, uln_ast, uln_bili) $ 20 ;
  if nmiss(bilibl, astbl, uln_ast, uln_bili) = 0 then do ;
    if 0<=bilibl and bilibl<=uln_bili and 0<=astbl and astbl<=uln_ast
      then return('NORMAL');
    else if (0<=bilibl and bilibl<= uln_bili and astbl>uln_ast>=0) or
      (uln_bili>=0 and uln_bili<bilibl and bilibl<=1.5*uln_bili)
      then return('MILD');
    else if (uln_bili>=0 and 1.5*uln_bili<bilibl and bilibl<=3*uln_bili)
      then return('MODERATE') ;
    else if (uln_bili>=0 and 3*uln_bili < bilibl )
      then return('SEVERE');
  end;
  else do ;
    return('MISSING') ;
  end;
endsub;
run;

```

Our library of user-defined functions also includes additional clinical calculators, such as body mass index (BMI), body surface area (BSA), and creatinine clearance (CrCl), among others. Code and documentation are available upon request.

CONCLUSION

In summary, user-defined functions are very practical and useful in performing complex and repeat computation in clinical study data processing and derivation. A set of validated functions can help improve computation accuracy and programming productivity.

REFERENCES

1. Previous eGFR Equations for Reference | National Institute of Diabetes and Digestive and Kidney Diseases
(<https://www.niddk.nih.gov/research-funding/research-programs/kidney-clinical-research-epidemiology/laboratory/glomerular-filtration-rate-equations/adults/previous>)
2. CKD-EPI Creatinine Equation (2021) | National Kidney Foundation
(<https://www.kidney.org/professionals/ckd-epi-creatinine-equation-2021>)
3. Optimize Your Fitness with Our Lean Body Mass Calculator | Shohoz Calculator
(<https://shohozcalculator.com/lean-body-mass-calculator>)