Web-based Supporting Materials for "Analysis of Censored Longitudinal Data with Skewness and a Terminal Event" by Xiao Su and Sheng Luo

1 Results of fitting the joint and reduced models under various assumptions of random errors.

Table 1: The means, standard errors (SE), p values, and 95% confidence interval (CI) of the parameters from the joint model JM_N assuming piecewise-constant baseline hazard function.

	MLE	SE	P	95% CI
Log of Viral Load				
Int	17.163	0.209	< 0.001	16.752, 17.574
$\sqrt{\text{CD4}}$	-0.395	0.006	< 0.001	-0.408, -0.383
Race:white	-0.270	0.188	0.151	-0.639, 0.099
Age	-0.374	0.065	< 0.001	-0.501, -0.247
Time	-0.329	0.005	< 0.001	-0.338, -0.320
σ_u^2	2.225	0.113	< 0.001	2.003, 2.447
σ_ϵ	2.207	0.016	< 0.001	2.176, 2.238
Time to Death				
$\sqrt{\text{BCD4}}$	-0.150	0.011	< 0.001	-0.173, -0.128
Race:white	0.031	0.129	0.810	-0.222, 0.284
Age	0.149	0.048	0.002	0.056, 0.242
γ	0.241	0.028	< 0.001	0.185, 0.296

Table 2: The means, standard errors (SE), p values, and 95% confidence interval (CI) of the parameters from the joint model RM_{SN} assuming piecewise-constant baseline hazard function.

	MLE	SE	P	95% CI
Log Viral Load				
Int	18.854	0.229	< 0.001	18.405, 19.303
$\sqrt{\text{CD4}}$	-0.404	0.006	0.000	-0.417, -0.392
Race:white	-0.252	0.182	0.167	-0.610, 0.106
Age	-0.383	0.064	< 0.001	-0.509, -0.258
Time	-0.338	0.005	< 0.001	-0.347, -0.329
σ_u^2	2.164	0.111	< 0.001	1.947, 2.382
σ_ϵ	2.772	0.083	< 0.001	2.609, 2.935
λ	-1.101	0.116	< 0.001	-1.329, -0.872
Time to Death				
$\sqrt{\text{BCD4}}$	-0.111	0.010	< 0.001	-0.131, -0.092
Race:white	-0.005	0.120	0.968	-0.240, 0.230
Age	0.134	0.045	0.003	0.047, 0.222

2 SAS program to fit the proposed skew-normal joint model (8) with piecewise-constant baseline hazard function

We include this code in a file named `SkewNormal.sas` to be called in SAS procedure `NLMIXED`.

```
*Skew normal PDF;
proc fcmp outlib=sasuser.funcs.math;
    function SkewnormalPDF(x,location,scale,shape);
    out=2*PDF('NORMAL',x,location,scale)*CDF('NORMAL',shape*(x-location)/scale);
    if out eq 0 then out=1e-15;
    return (out);
    endsub;
run;

*load the dll file for owenT function;
proc proto  package=sasuser.funcs.math;
    link 'U:\Research\LuoGroupProjects\Su_Xiao\code_library\C\math\OwenT\OwenT.dll';
    double tha ( double h1, double h2, double a1, double a2);
run;

*wrapper for owenT;
options cmplib=sasuser.funcs;
run;
proc fcmp  outlib=sasuser.funcs.math;
    function OwenT(h,shape);
    return (tha (h,1.0,shape,1.0));
    endsub;
run;

*Skew normal CDF;
proc fcmp outlib=sasuser.funcs.math;
    function SkewnormalCDF(x,location,scale,shape);
    out=CDF('NORMAL',(x-location)/scale)-2*OwenT((x-location)/scale,shape);
    if out<0 then out=abs(out) ;
    if out=0 then out=1e-15;
    if out>1 then out=1;
    return (out);
    endsub;
run;
}
```

Below is the main code to fit the proposed skew-normal joint model (8) with piecewise-constant baseline hazard function by using SAS procedure `NLMIXED`.

```
/* Invoke proc nlmixed using Gaussian Quadrature */
proc nlmixed data=CLreal.cleandata  maxit=5000 gtol=1e-15 noad qpoints=50 ;
  /* define the array */
  /* cut points for the intervals of terminal event; */
      array r_D{10} r_D1-r_D10;
  /* piecewise constant function for death; */
      array tau_D{11} tau_D1-tau_D11;
  /* specify the cut points for the intervals; */
    tau_D[1]=0;tau_D[2]=0.4;tau_D[3]=0.6;tau_D[4]=0.7;tau_D[5]=0.8;tau_D[6]=0.9;
    tau_D[7]=1;tau_D[8]=1.1;tau_D[9]=1.4;tau_D[10]=1.8;tau_D[11]=2.3;
  /* set boundaries for parameters; */
     bounds r_D1 r_D2 r_D3 r_D4 r_D5 r_D6 r_D7 r_D8 r_D9 r_D10>0, sigma2u>0,sigma2e>0 ;
  /* initialize parameters; */
      parms r_D1=0.1,r_D2=0.2,r_D3=0.1,r_D4=0.1, r_D5=0.1, r_D6=0.1, r_D7=0.1, r_D8=0.1,
```

```
      r_D9=0.1,  r_D10=0.1,coeffCD4S=-0.2,coeffwhiteS=0.5,coeffageS=-0.1,
      coeffCD4L=-0.1 ,coeffwhiteL=2 ,coeffageL=-0.36,coefftimeL=-0.2,
      intercept=10,sigma2u=1,gamma=0.2,shape=-1,sigma2e=1;
     *Calculate the cumulative hazard for relapse and death processes;
     if newvisitID eq 1 then do;
         G_D=0;
         do i=1 to 10;
             if i<IndicatorD then G_D=G_D+r_D[i]*(tau_D[i+1]-tau_D[i]);
             if i=IndicatorD then G_D=G_D+(TimeTerminal-tau_D[i])*r_D[i];
             end;
         etaS=coeffCD4S*baselineCD4count10+coeffwhiteS*white+coeffageS*baselineage10+u*gamma;
         r=r_D[IndicatorD]*exp(etaS);
         S= exp(-exp(etaS)*G_D);
         if DeathCensor eq 1 then LSur=r*S;
         if DeathCensor eq 0 then LSur=S;
         end;
     else LSur=1;
     etaL=Intercept+coeffCD4L*CD4count10+coeffwhiteL*white+coeffageL*baselineage10+coefftimeL*timetoenroll+u;
     if Vloadcensor eq 0 then Llong=SkewnormalPDF(logvload,etaL,sigma2e,shape);
     if Vloadcensor eq 1 then Llong=SkewnormalCDF(logvload,etaL,sigma2e,shape);
     ll=log(LSur*Llong);
         *specify random effect;
     random u~normal([0],[sigma2u]) subject=newid;
     model TimeTerminal~general(ll);
     ods output ParameterEstimates=Output.PSHat;
     ods output  FitStatistics=Output.PSFit;
run;
}
```