**Supplementary document 1: Stata and R codes of all four statistical methods applied in estimating age-specific paediatric RIs (**Monsurul Hoq Susan Donath,Paul Monagle, John B. Carlin Comparison of statistical methods for estimating continuous paediatric reference intervals: a simulation study)

***1-A: Stata Code for Royston’s method***

**\* Developed by Monsurul Hoq for the purpose of the comparing four methods for estimating continuous RI.**

**\* The following program is applicable for estimating RIs (2.5th - 97.5th) using age and sex (where 1 = Male and 0 = Female).**

**\* A brief discussion of the method is include in the masnucript.**

**\* Further details including "xrigls" and "xriml" can be found in Royston *et al*, JRSS 1998. 161(1): p. 79-101.**

**clear**

**set more off**

**cap log close**

**version 15.1**

**cap program drop rri**

**program rri**

 **\* step 1: identify the best fitting model for mean and SD**

 **\* For the entire code 1=analyte, 2=age and 3=sex**

 **xrigls `1' `2', nograph fp(m:df 4, s:df 4) centile(2.5 97.5) // xrigls is a user-defined package in Stata develped by P. Royston (JRSS 1998. 161(1): p. 79-101)**

 **local m\_fp `r(mpow)' // store the polynomial powers of age for the mean**

 **local s\_fp `r(spow)' // store the polynomial powers of age for the standard deviation**

 **if "`m\_fp'"=="" { // If no or linear association with age and constant SD**

 **xriml `1' `2', dist(n) se centile(2.5 97.5) nograph noscaling // xriml is a user-defined package in Stata develped by P. Royston (JRSS 1998. 161(1): p. 79-101)**

 **}**

 **else {**

 **\* Step 2: Assess the difference in mean response by sex for the model**

 **\* using the best fitting power variables of age in the previous step**

 **if strlen("`m\_fp'") <= 2 { // if the 1-degree polynomials were identified for the mean**

 **fp <`2'>, replace powers(`m\_fp') dim(1): reg `1' `3' <`2'>**

 **mat b = e(b) // store the estimates of b coefficients**

 **mat se = e(V) // store the variance-covarience matrix**

 **}**

 **else if strlen("`m\_fp'") > 2 { // if the 2-degree polynomials were identified for the mean**

 **fp <`2'>, replace powers(`m\_fp') dim(2): reg `1' `3' <`2'>**

 **mat b = e(b)**

 **mat se = e(V)**

 **}**

 **estimate store a // store the findings from the regression y = a + b1\*age+b2\*sex**

 **\* store the p-value corresponding to the b-coefficient of sex**

 **local t\_sex = b[1,1] / sqrt(se[1,1])**

 **local p\_sex = tprob(e(N)-2,`t\_sex')**

 **di `p\_pex'**

 **\* Step 3: Is there an interaction between the age (selected degree of FP models) and sex ?**

 **\*(so far the power variables selected in the -mfpi- were consistent with -mfp-)**

 **reg `1' c.`2'\_?##i.`3' // here `2'\_? will recall the power variables of age i.e `m\_fp'**

 **estimate store b // store the findings from the regression y = a + b1\*age+b2\*sex+b3\*age\*sex**

 **lrtest a b, stats // compare the two models with or without interaction term**

 **local p\_int `r(p)' // Ho: No difference in deviance between the model significant. Store the p-value**

 **di `p\_int'**

 **if `p\_sex' > 0.05 & `p\_int' > 0.05 { // if both sex and interaction with sex is insignificant**

 **if "`s\_fp'"=="" { // if SD is constant**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling // normal distribution**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling // exponential normal distribution**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling // modulus exponential normal distribution**

 **local dev\_men = 2\*e(ll)**

 **\*Comparison of deviance between the 3 modeles**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **\* Selection of the final model**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **}**

 **else { // if s is a function of age**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **}**

 **}**

 **else if `p\_sex' < 0.05 & `p\_int' > 0.05 { // there is a difference by sex but not seperate by sex**

 **if "`s\_fp'"=="" { // if S is constant**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **}**

 **else { // if s is a function of age**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling covar(m:`3', s:`3')**

 **}**

 **}**

 **}**

 **else if `p\_int' < 0.05 { // seperately by sex**

 **\*since its not possible to include interaction term in the model, RIs are estimated seperately by sex**

 **if "`s\_fp'"=="" { // if S is consistent**

 **tempfile male female**

 **preserve**

 **keep if `3'==1**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **save `male', replace**

 **restore**

 **keep if `3'==0**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **append using `male'**

 **}**

 **else { // if s is a function of age**

 **tempfile male female**

 **preserve**

 **keep if `3'==1**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **save `male', replace**

 **restore**

 **keep if `3'==0**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_n = 2\*e(ll)**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_en = 2\*e(ll)**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **local dev\_men = 2\*e(ll)**

 **local diff1 = abs(`dev\_n' - `dev\_en')**

 **local diff2 = abs(`dev\_en' - `dev\_men')**

 **local p1 = 1-chi2(1,`diff1')**

 **local p2 = 1-chi2(1,`diff2')**

 **if `p1' > 0.05 {**

 **xriml `1' `2', dist(n) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else if (`p1' <.05) & (`p2' > .05) {**

 **xriml `1' `2', dist(en) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **else {**

 **xriml `1' `2', dist(men) fp(m:`m\_fp', s: `s\_fp') se centile(2.5 97.5) nograph noscaling**

 **}**

 **append using `male'**

 **}**

 **}**

 **}**

**end**

***B: Stata codes for Hoq et al’s method***

**\* Developed by Monsurul Hoq for the purpose of the comparing four methods for estimating continuous RI.**

**\* The following program is applicable for estimating RIs (2.5th - 97.5th) using age and sex (where 1 = Male and 0 = Female).**

**\* A brief discussion of the method is include in the masnucript.**

**\* Further details can be found in Hoq *et al*, Clin Chem 2019 65(10).**

**clear**

**set more off**

**cap log close**

**version 15.1**

**cap program drop ri**

**program ri**

 **\* Regress the mean of y on age and sex to identify the best fitting power variable of age**

 **\* For the entire code 1=analyte, 2=age and 3=sex**

 **mfp, center(no, `2':mean): reg `1' `2' `3'**

 **local pw `e(fp\_k1)' // store selected powers of age for the mean**

 **\* Is the difference in mean response by sex significant ?**

 **mat b = e(b) // store the estimates of b coefficients**

 **mat se = e(V) // store the variance-covarience matrix**

 **local col = `e(rank)'-1 // to understand the position of the sex variable in the model**

 **\* store the p-value corresponding to the b-coefficient of sex**

 **local t\_sex = b[1,`col'] / sqrt(se[`col',`col'])**

 **local p\_sex = tprob(e(N)-2,`t\_sex')**

 **\* Is there an interaction between the age (selected powers) and sex ?**

 **\* For a linear model**

 **if "`pw'"=="1" { // `pw' stores the power variables as string**

 **mfpi reg `1' `2', showmodel with(`3') linear(`2')**

 **local p\_int `r(Plin)' // Ho: No difference in deviance between the model significant. Store the p-value**

 **}**

 **\* For a 1-Degree Fractional Polynomial model**

 **else if strlen("`pw'") <= 2 {**

 **mfpi reg `1' `2', showmodel with(`3') fp1(`2')**

 **local p\_int `r(Pfp1)'**

 **}**

 **\* For a 2-Degree Fractional Polynomial model**

 **else if strlen("`pw'") > 2 {**

 **mfpi reg `1' `2', showmodel with(`3') fp2(`2')**

 **local p\_int `r(Pfp2)'**

 **}**

 **if `p\_sex' > 0.05 & `p\_int' > 0.05 { // both sex and interaction with sex is insignificant**

 **fp <`2'>, fp("`pw'") replace all : reg `1' <`2'> // This will generate the power variables of age (age\_?)**

 **\*estimate the RIs (i.e. upper and lower limits) applying quantile regression where power variables of age are the covriates**

 **qreg `1' `2'\_?, quantile(2.5) vce(,bof)**

 **predict `1'\_ll**

 **predict `1'\_ll\_se1, stdp**

 **qreg `1' `2'\_? , quantile(97.5) vce(,bof)**

 **predict `1'\_ul**

 **predict `1'\_ul\_se1, stdp**

 **}**

 **else if `p\_sex' < 0.05 & `p\_int' > 0.05 { // there is a difference by sex but not seperate by sex**

 **fp <`2'>, fp("`pw'") replace all : reg `1' i.`3' <`2'>**

 **qreg `1' `2'\_? i.`3', quantile(2.5) vce(,bof)**

 **predict `1'\_ll**

 **predict `1'\_ll\_se1, stdp**

 **qreg `1' `2'\_? i.`3', quantile(97.5) vce(,bof)**

 **predict `1'\_ul**

 **predict `1'\_ul\_se1, stdp**

 **}**

 **else if `p\_int' < 0.05 { // seperate by sex**

 **fp <`2'>, fp("`pw'") replace all : reg `1' i.`3' <`2'>**

 **qreg `1' c.`2'\_?##i.`3', quantile(2.5) vce(,bof)**

 **predict `1'\_ll**

 **predict `1'\_ll\_se1, stdp**

 **qreg `1' c.`2'\_?##i.`3', quantile(97.5) vce(,bof)**

 **predict `1'\_ul**

 **predict `1'\_ul\_se1, stdp**

 **}**

 **end**

***1-C: State code for Cole’s LMS method***

**\* The following code from ZWANG was update by Monsurul Hoq to select the EDF based on a cross-validation.**

**\*! version 0.1 ZWANG Nov 1998, Menzies School of Health Research**

**\*! version 0.2 M.S. Pearce Nov 1998, Child Health, University of Newcastle**

**\* upon Tyne**

**cap program drop colelms2\_cx**

**program define colelms2\_cx**

 **version 5.0**

 **local varlist "req ex min(1) max(1)"**

 **#delimit ;**

 **local options "AGEgroup(string) SEX(string) df(string)**

 **NOSmooth CENtile(string) save(string) ";**

 **#delimit cr**

 **parse "`\*'"**

 **parse "`varlist'", parse(" ")**

 **tempvar xlog xrec mg mh ma sg sa sh A B sd sdlog sdr**

 **if "`centile'"=="" {local centile "2.5 10 25 50 75 90 97.5"}**

 **gen `xlog'=ln(`1')**

 **gen `xrec'=1/`1'**

 **local GM "`1'"**

 **if "`save'"==""{local save "LMStemp"}**

 **if "`sex'"!=""{**

 **# delimit ;**

 **collapse (mean) "`1'" `xlog' `xrec' (sd) sd="`1'" `sdlog'=`xlog'**

 **`sdr'=`xrec' (count) num=`1', by(`sex' `agegroup');**

 **# delimit cr**

 **}**

 **if "`sex'"==""{**

 **# delimit ;**

 **collapse (mean) "`1'" `xlog' `xrec' (sd) sd="`1'" `sdlog'=`xlog'**

 **`sdr'=`xrec' (count) num=`1', by(`agegroup');**

 **# delimit cr**

 **}**

 **gen `mg'=exp(`xlog')**

 **gen `mh'=1/`xrec'**

 **gen `ma'=`1'**

 **gen `sg'=`sdlog'**

 **gen `sa' =sd/`mg'**

 **gen `sh'=`sdr'\*`mg'**

 **gen `A'=ln(`sa' /`sh')**

 **gen `B'=ln((`sa' \*`sh')/(`sg'^2))**

 **qui {**

 **gen L=-`A'/(2\*`B')**

 **gen S=`sg'\*exp(`A'\*L/4)**

 **gen M=`mg'+(`ma'-`mh')\*L/2+(`ma'-2\*`mg'+`mg')\*(L^2)/2**

 **\* Cross-validation to select the optimum EDF for fitting a model for mu**

 **forvalue i = 1/18 {**

 **forvalue o = 0/1 {**

 **forvalue b = 2/6 {**

 **\* Drop each interage age at a time and estimate the model using the reamining age and mean for df = 2 - 6**

 **gam M `agegroup' if `sex'==`o' & `agegroup'!=`i', df(`b')**

 **rename GAM\_mu MS1m`i'`o'`b'**

 **}**

 **}**

 **}**

 **gen diff2 = .**

 **gen diff3 = .**

 **gen diff4 = .**

 **gen diff5 = .**

 **gen diff6 = .**

 **\* Calcuate the error in predicting the new data**

 **forvalue i = 2/17 {**

 **forvalue o = 0/1 {**

 **forvalue b = 2/6 {**

 **local j = `i'-1**

 **local k = `i'+1**

 **sum MS1m`i'`o'`b' if inlist(age, `j' , `k') & sex==`o'**

 **replace MS1m`i'`o'`b' = r(mean) if age==`i' & sex==`o'**

 **replace diff`b' = M - MS1m`i'`o'`b' if age==`i' & sex==`o'**

 **}**

 **}**

 **}**

 **\* identify the EDF between 2 and 6 with minimum error in predicting the mean**

 **forvalue i = 2/6 {**

 **gen diff\_s`i' = sqrt((diff`i')^2)**

 **sum diff\_s`i'**

 **}**

 **preserve**

 **collapse (mean) diff\_s?**

 **gen id=1**

 **reshape long diff\_s, i(id)**

 **sort diff\_s**

 **local df = \_j[1]**

 **di "`df'"**

 **restore**

 **\* use the the identified EDF for calculating the centiles**

 **if "`nosmooth'"==""{**

 **if "`sex'"==""{**

 **gam L `agegroup', df(`df')**

 **gen LS=GAM**

 **gam M `agegroup', df(`df')**

 **gen MS=GAM**

 **gam S `agegroup', df(`df')**

 **gen SS=GAM**

 **}**

 **if "`sex'"!=""{**

 **quietly summ `sex'**

 **local one=\_result(5)**

 **local two=\_result(6)**

 **gam L `agegroup' if `sex'==`one', df(`df')**

 **gen LS=GAM if `sex'==`one'**

 **gam L `agegroup' if `sex'==`two', df(`df')**

 **replace LS=GAM if `sex'==`two'**

 **gam M `agegroup' if `sex'==`one', df(`df')**

 **gen MS=GAM if `sex'==`one'**

 **gam M `agegroup' if `sex'==`two', df(`df')**

 **replace MS=GAM if `sex'==`two'**

 **gam S `agegroup' if `sex'==`one', df(`df')**

 **gen SS=GAM if `sex'==`one'**

 **gam S `agegroup' if `sex'==`two', df(`df')**

 **replace SS=GAM if `sex'==`two'**

 **}**

 **lab var LS "L smoothed"**

 **lab var MS "M smoothed"**

 **lab var SS "S smoothed"**

 **drop s\_ r\_ e\_ GAM**

 **parse "`centile'", parse(" ")**

 **while "`1'" !="" {**

 **local tcent = `1'/100**

 **local new = subinstr("`1'", ".", "x",.)**

 **local Z`new' = invnorm(`tcent')**

 **gen C`new' = MS\*((1+LS\*SS\*`Z`new'')^(1/LS))**

 **mac shift**

 **}**

 **}**

 **}**

 **end**

***1- D: R code for GAMLSS***

The following steps were applied to estimate RIs using GAMLSS method.

rm(list=ls())

#setwd("Y:/PhD/Simulation/ALP")

**#call the packages**

library(gamlss)

library(haven)

library(MASS)

source("extract\_centiles.R")

SS=c(100,200,400,1000)

for (k in 1:4) {

 for (i in 1:1000) {

 S=Sys.time()

 tryCatch(

 {

 print(i)

**# reading the data**

 datafile1 <-read\_stata(paste0("Data/",SS[k],"/size",SS[k],"\_sim",i,".dta"))

 alp\_all <- subset(datafile1, select=c(age,sex,alp))

 alp\_all0 <- na.omit(alp\_all)

 alp\_all0$alp[alp\_all0$alp<1]<-1

 alp\_all0$id<-1:nrow(alp\_all0)

**# finding the hyper parameters for mean and SD i.e. edf for model including age and sex**

 m1 <- quote(gamlss(alp~cs(age,df=p[1])+as.factor(sex), data=alp\_all0, family=BCT(), control=gamlss.control(trace=FALSE)))

 op <-find.hyper(model=m1, par=c(3), lower=c(1), steps=c(.1), k=log(SS[k]))

 m2 <- gamlss(alp~cs(age,df=op$par[1])+as.factor(sex),data=alp\_all0, family=BCT())

**# assess the difference in mean by sex**

 m3<-stepGAIC.VR(m2)

 m3$anova

**# assess the interaction between age and sex**

 m4<-stepGAIC.VR(m2, scope=list(lower=~cs(age,df=op$par[1]), upper=~(cs(age,df=op$par[1])+as.factor(sex))^2))

 m4$anova

**# assess the difference in SD by sex**

 m5<-stepGAIC.VR(m4, what="sigma",scope=~(cs(age,df=op$par[1])+as.factor(sex)))

 m5$anova

**# Find the hyper parameter for mean and SD for the selected model**

 m6 <- quote(gamlss(m4$mu.formula,sigma.fo=m5$sigma.formula, data=alp\_all0, family=BCT(), control=gamlss.control(trace=FALSE)))

 op <-find.hyper(model=m6, par=c(3,3), lower=c(1,1), steps=c(.1,.1), k=log(SS[k]))

 m7 <- gamlss(m4$mu.formula,sigma.fo=m5$sigma.formula,data=alp\_all0, family=BCT())

**# calculate the centile**

 o1<-extract\_centiles(m7,xvar=alp\_all0$age,idvar=alp\_all0$id,groupvar = alp\_all0$sex, col.cent=1,

 cent=c(2.5,97.5),

 lty.centiles=c(3,3, 2,2,1,2, 2,3, 3),

 lwd.cent=c(1,1,1.5,1.5,2,1.5,1.5,1,1),

 ylab = "ALP",

 xlab = "Age (years)",

 legend = F,

 xlim = c(min(alp\_all0$age),18),

 save=F,

 las=1)

 results<-read.csv("Coordinates for selected percentiles.csv" )

 write.csv(results,paste0("Results/",SS[k],"/methods5\_size",SS[k],"\_sim",i,".csv"), row.names = FALSE)

 }, error=function(e){})

 }

}