```
*  Potential Mast Production Module: Formulated for LANDIS output
*  STATA (V.6) syntax

*
*  By: Neal Sullivan - Forestry Department, University of Missouri
*              www.snr.missouri.edu/~sullivan

*
*           Last Edited:      3/01/02 10:21     NHS
*
* Begin Commands:

      drop _all
      set memory 400m
      use filename.dta, replace

*     keep if stand<20
      set more 1

* To run: using a generic ageclass,
*     create std_age (Simulation Age) = agep10 (Landis decade of simulation)

      gen std_age=agep90

* *

* Est. number of dominant + codominant oak trees per cell by ageclass

      local tpc_sl = -0.099               /* slope of age tpc relationship*/

      local tpc_con = 24.1                /* intercept of age tpc relationship*/

      local tpcsdsl = -0.08               /* slope of age tpc-sd relationship*/

      local tpcsdco = 16                  /* intercept of age tpc-sd relationship*/

* number of cell canopy oak trees (tpc):
*           random selection from normal distribution by age of dominants

      gen tpc=(`tpc_sl'*std_age+`tpc_con')+
            (`tpcsdsl'*std_age+`tpcsdco')*invnorm(uniform())


* Influence of Landtype on the number of canopy oak trees >12.7 D   MOFEP Based

      /*          Mean =      0.887                   */
      /*          multiplier                          */
      /*    1     1.119                               */
      /*    2     0.993                               */
      /*    3     1.164                               */
      /*    4     0.856                               */
      /*    5     0.867                               */
      /*    6     1.105                               */
      /*    7     0.895                               */

      replace     tpc   =tpc  *     1.119 if landunit==1
      replace     tpc   =tpc  *     0.993 if landunit==2
```

```
      replace     tpc   =tpc  *     1.164 if landunit==3
      replace     tpc   =tpc  *     0.856 if landunit==4
      replace     tpc   =tpc  *     0.867 if landunit==5
      replace     tpc   =tpc  *     1.105 if landunit==6
      replace     tpc   =tpc  *     0.895 if landunit==7

      gen int cotpc  = round(tpc  ,0)
            /* number of canopy trees rounded to whole number */

            drop tpc
```

*** Estimate of cell-stand mean diameter and distribution by age of dominants**

```
* from FIA data, mean diameter per ageclass

* Assign a random diameter to 40 trees for each cell for each iteration


      for new tr_1-tr_40: gen X = 0.5443953* std_age - (std_age*std_age*0.0025517) +
                                  (std_age*std_age*std_age*4.11e-06)

      /* Apply variability */

      for var tr_1-tr_40: replace X = X + (0 + (0.1670191* std_age-
                          (std_age*std_age*0.000432)) * invnorm(uniform()))

      /* replace diameter with 0 if very small */

      for var tr_1-tr_40: replace X = 0 if X < 1.5



/*     Variation in Oak component (prop.)    Stand Structure  (D)       */

/*    LANDIS            Affect on                    */
/*    Type  ELT         Diameter                     */
/*    1     17          1.001                        */
/*    2     18          1.082                        */
/*    3     11          1.164                        */
/*    4     19          0.87                         */
/*    5     5,7         0.956                        */
/*    6     20,21       0.964                        */
/*    7     other       0.965                        */


      for var  tr_1-tr_40: replace X =  X  *    1.001 if landunit==1
      for var  tr_1-tr_40: replace X =  X  *    1.082 if landunit==2
      for var  tr_1-tr_40: replace X =  X  *    1.164 if landunit==3
      for var  tr_1-tr_40: replace X =  X  *    0.87  if landunit==4
      for var  tr_1-tr_40: replace X =  X  *    0.956 if landunit==5
      for var  tr_1-tr_40: replace X =  X  *    0.964 if landunit==6
      for var  tr_1-tr_40: replace X =  X  *    0.965 if landunit==7

* Make diameter 0 if tree-N > canopy trees per cell


      replace     tr_1  = 0 if      cotpc < 1
```

```
replace     tr_2  = 0 if      cotpc < 2
.
.
.
replace     tr_40 = 0 if      cotpc < 40
```

```
* Factor regional variation in ratio of red oak and white oak groups
```

```
      * by ageclass:    fraction of oaks that are red oak = -0.0027066*(ageclass) +
                                                            0.6286, R2 =.86 for FIA

      * adjustment for Oregon County:  -0.0000179 * ageclass^2 + .644


      gen cfro =        -0.0000179*(std_age)^2 + 0.644

* Adjust for landis land type (unit)


      /*    Land type   MOFEP AVG   Adj.  Var. (SD)   */

      /*          0.7111                    */
      /*    1     0.741 1.04        */
      /*    2     0.734 1.04        */
      /*    3     0.831 1.17        */
      /*    4     0.647 0.91  0.05  */
      /*    5     0.593 0.83        */
      /*    6     0.732 1.03        */
      /*    7     0.700 0.98        */

      gen mncfro =                  1.04  * cfro

      replace mncfro =        1.04  * cfro            if landunit==1

      replace mncfro =        1.17  * cfro            if landunit==2

      replace mncfro =        0.91  * cfro            if landunit==3

      replace mncfro =        0.83  * cfro            if landunit==4

      replace mncfro =        1.03  * cfro            if landunit==5

      replace mncfro =        0.98  * cfro            if landunit==6

*  Add random variability to Fraction red oak group

      replace mncfro = mncfro + 0.05 * invnorm(uniform())
```

```
* Randomly assign trees to ro and wo species groups

      /* create a species code for each tree on each cell */

      for new sp_1 -sp_40:                gen str2 X = "WO"

      for new temp1-temp40: gen X = (uniform())

      /* randomly assign tree to sp group */

      replace  sp_1           ="RO" if    temp1 < mncfro
      replace  sp_2           ="RO" if    temp2 < mncfro
      .
      .
      .
      replace  sp_40          ="RO" if    temp40      < mncfro


      drop temp1-temp40
```

```
* Adjust diameter for different growth rates for subgenus's
```

```
*
*     model of multiplier to get WO scaler

*
*           adjust =    -0.0010492  (age) +     1.010843

*
      replace tr_1  = (std_age * -0.0010492 + 1.011)   * tr_1 if sp_1 =="WO"
      replace tr_2 = (std_age* -0.0010492 + 1.011)     * tr_2 if sp_2 =="WO"
      .
      .
      .
      replace tr_40 = (std_age* -0.0010492 + 1.011)    * tr_40 if sp_40 =="WO"

* Set  diameter to 0 if very small

      for var tr_1-tr_40: replace X = 0 if X < 1.5
```

```
*  Tree diameter and mast production (index) relationship----
*                 generates a 0-1 index of relative mast production
```

```
                                  /*          Coeff. (0-1)DBH       */
                                  /*                                */
      local rod  = -3.361032      /*                -0.22407  D     */
      local rod2 = 0.1279035      /*  Scaler        0.00853   D2    */
      local rod3 = -0.0017115     /*for desired     -1.14E-04 D3    */
      local rod4 = 0.000007515    /*  range         5.01E-07  D4    */
      local rodc = 28.65          /* Maximum =15    1.91000 interc. */

                                  /* Similar to range of kg/tree    */
      local wod =-1.4629095       /*    0     -0.09753      D       */
      local wod2 = 0.0567975      /*    0     0.00379       D2     */
```

```
      local wod3 = -0.000705        /*    0     -4.70E-05     D3     */
      local wod4 = 0.000002835      /*    0     1.89E-07      D4     */
      local wodc = 11.85            /*    0     0.79000     interc.  */

      gen rms_1 =tr_1*`rod'+(tr_1^2)*`rod2'+(tr_1^3)*
                        `rod3'+ (tr_1^4)*`rod4'+`rodc' if sp_1=="RO"
      gen rms_2=tr_2*`rod'+(tr_2^2)*`rod2'+(tr_2^3)*
                        `rod3'+ (tr_2^4)*`rod4'+`rodc'  if sp_2=="RO"
      gen rms_3=tr_3*`rod'+(tr_3^2)*`rod2'+(tr_3^3)*
                        `rod3'+ (tr_3^4)*`rod4'+`rodc'  if sp_3=="RO"
      .
      .
      .
      gen rms_40=tr_40*`rod'+(tr_40^2)*`rod2'+(tr_40^3)*
                        `rod3'+ (tr_40^4)*`rod4'+`rodc' if sp_40=="RO"


      gen wms_1=tr_1*`wod'+(tr_1^2)*`wod2'+(tr_1^3)*
                        `wod3'+ (tr_1^4)*`wod4'+`wodc'if sp_1=="WO"
      gen wms_2=tr_2*`wod'+(tr_2^2)*`wod2'+(tr_2^3)*
                        `wod3'+ (tr_2^4)*`wod4'+`wodc'if sp_2=="WO"
      gen wms_3=tr_3*`wod'+(tr_3^2)*`wod2'+(tr_3^3)*
                        `wod3'+ (tr_3^4)*`wod4'+`wodc'if sp_3=="WO"
      .
      .
      .
      gen wms_40=tr_40*`wod'+(tr_40^2)*`wod2'+(tr_40^3)*
                        `wod3'+ (tr_40^4)*`wod4'+`wodc'if sp_40=="WO"

* Correct for artifact of mast = f(D)*function for very large trees


      replace rms_1  = 5 if   tr_1   > 91 &  sp_1 =="RO"
      replace rms_2  = 5 if   tr_2   > 91 &  sp_2 =="RO"
      .
      .
      .
      replace rms_40    = 5 if      tr_40 > 91 &  sp_40 =="RO"
```

```
* Factor per tree inherent variability in acorn production  (randomly assigned)

*           Fraction of population        /* proportion of population   */
*     Non-producers           0.15        /*          0.10              */
*     Low Prod          0.30              /*          0.30              */
*     Mod Prod          0.40              /*          0.45              */
*     High Prod         0.15              /*          0.15              */
*
*     /* macros assigning class thresholds */

*                            relative  production rate

*     No prod           (gt)  0.00
*     Low Prod    (gt)  0.15              0.30
*     Mod Prod    (gt)  0.45              0.75
*     High Prod   (gt)  0.85              1.00

      for new pr_1-pr_40: gen X = 0

      for new temp1-temp40: gen X = (uniform())

      replace pr_1 =     0.3    if temp1   >  0.15
      replace pr_2 =     0.3    if temp2    >  0.15
      .
      .
      replace pr_40 =    0.3    if temp40  >  0.15

      replace pr_1 =    0.7    if temp1    >  0.45
      replace pr_2 =    0.7    if temp2    >  0.45
      .
      .
      replace pr_40 =    0.7     if temp40  >  0.45

      replace pr_1 =    1    if  temp1     >  0.85
      replace pr_2 =    1    if  temp2     >  0.85
      .
      .
      replace pr_40 =    1    if temp40    >  0.85


      drop temp*


*  Scale per tree mast production for red oak group by randomly assigned but subjectve
productivity classes


      replace rms_1  = pr_1  *  rms_1
      replace rms_2  = pr_2  *  rms_2
      .
      .
      replace rms_40 = pr_40 *  rms_40

*  Re-calculate for assignment of inherent productivity  to white oak group
```

```
      for new temp1-temp40: gen X = (uniform())

      replace pr_1 =   0.30  if temp1  >  0.15
      replace pr_2 =   0.30  if temp2  >  0.15
      .
      .
      replace pr_40 =   0.30  if temp40  >  0.15

      replace pr_1 =  .70  if temp1  > 0.45

      replace pr_2 =  .70  if temp2  > 0.45
      .
      .
      replace pr_40 =  .70  if temp40  > 0.45

      replace pr_1 = 1 if temp1  >   0.85
      replace pr_2 = 1 if temp2  >   0.85
      .
      .
      replace pr_40 = 1 if temp40  >   0.85

      drop temp*


*  Scale per tree mast production for white oak group by randomly
*                       assigned but subjectve productivity classes

      replace wms_1       = pr_1 *  wms_1
      replace wms_2       = pr_2 *  wms_2
      .
      .
      replace wms_40      = pr_40 *  wms_40

* Set mast production at 0 if trees less that 20 cm in diameter

      replace rms_1     = 0 if      tr_1< 20 & sp_1=="RO"
      replace rms_2     = 0 if      tr_2< 20 & sp_2=="RO"
      .
      .
      replace rms_40    = 0 if      tr_40 < 20 & sp_40=="RO"

      replace wms_1     = 0 if      tr_1 < 20 & sp_1=="WO"
      replace wms_2     = 0 if      tr_2 < 20 & sp_2=="WO"
      .
      .
      replace wms_40    = 0 if      tr_40 < 20 & sp_40=="WO"

* drop per tree diameter information  for trees 2-40 ,
*                             tree 1 retained for model evaluation
      drop tr_2-tr_40

* drop per tree inherent productivity information for trees 2-40 ,
*                             tree 1 retained for model evaluation
      drop    pr_2-pr_40
```

```
*  Generate Mast production indices

*  Cell totals for red oak subgenus

      gen  rmst =rms_1 if sp_1=="RO"

      replace rmst =rmst +rms_2 if sp_2=="RO"
      replace rmst =rmst +rms_3 if sp_3=="RO"
      .
      .
      replace rmst =rmst +rms_40 if sp_40=="RO"

*  Cell totals for white oak subgenus

      gen  wmst =wms_1 if sp_1 =="WO"
      replace wmst =wmst +wms_2  if sp_2 =="WO"
      .
      .
      replace wmst =wmst +wms_40  if sp_40 =="WO"

* drop per tree level mast production trees 2-40,tree 1 retained for model evaluation

            drop   sp_2-sp_40
            drop    rms_2-rms_40
            drop    wms_2-wms_40

* Set cell subgenus mast production to zero if missing

            replace  rmst = 0 if rmst ==.
            replace  wmst = 0 if wmst ==.


* Generate overall cell mast production value


      gen mst = rmst+wmst

      gen kg = mst*10000/900 /* convert to scale similar to kg per ha 30 m cell */


*     Age of simulation used:
*                       resultsfilename.dta
```