

Testing Plotting Functions

2023-03-20

TO DO: - add CI's

Prep

Let's load a couple random-ish gamm models and the dataframe they're built on

```
## GAMLSS-RS iteration 1: Global Deviance = 1218118
## GAMLSS-RS iteration 2: Global Deviance = 1218033
## GAMLSS-RS iteration 3: Global Deviance = 1218022
## GAMLSS-RS iteration 4: Global Deviance = 1218017
## GAMLSS-RS iteration 5: Global Deviance = 1218017
## GAMLSS-RS iteration 6: Global Deviance = 1218017
## GAMLSS-RS iteration 7: Global Deviance = 1218017
## GAMLSS-RS iteration 8: Global Deviance = 1218017
## GAMLSS-RS iteration 9: Global Deviance = 1218017
## GAMLSS-RS iteration 10: Global Deviance = 1218017
## GAMLSS-RS iteration 11: Global Deviance = 1218017
## GAMLSS-RS iteration 12: Global Deviance = 1218017
## GAMLSS-RS iteration 13: Global Deviance = 1218017
```

Call the script holding the plot functions we want to test:

```
source("plotting_functions.R")
```

Test

Set-up Functions

These functions are called from the plotting functions.

sim.data() - takes the dataframe you built your GAMLSS model on and creates a 2 new dfs with simulated data (male vs female participants) across the age-range, assigning fs_version and study values to whatever is most common in the original df. Expects input df to have log_age, fs_version, and study. Also preps x-axis labels and defines which centiles you'll be plotting.

Returns object **sim**, which is a list of objects.

```
sim <- sim.data(cn_df)
names(sim)

## [1] "ageRange"           "dataToPredictM"      "dataToPredictF"
## [4] "tickMarks_log"      "tickLabels_log"      "tickMarks_unscaled"
## [7] "tickLabels_unscaled" "desiredCentiles"
```

`centile_predict()` - predicts centiles based on df simulated by `sim.data()` using the `predictAll()` and `qG()` functions. Calculates centiles, 50th centile peak values, and age at peaks separately on male and female dfs and returns each, as well as an averaged effect across sexes.

Takes GAMLSS model obj and objects returned from `sim.data()`. Returns object `pred`, which is a list of objects.

```
pred <- centile_predict(sGMV.re, sim$dataToPredictM, sim$dataToPredictF, sim$ageRange, sim$desiredCenti
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

```
names(pred)
```

```
## [1] "fanCentiles"    "fanCentiles_M"  "fanCentiles_F"  "peak"  
## [5] "peak_age"       "peak_M"        "peak_age_M"    "peak_F"  
## [9] "peak_age_F"     "M_mu"         "M_sigma"      "M_nu"  
## [13] "F_mu"          "F_sigma"      "F_nu"
```

`centile_predict.corrected()` - same as `centile_predict` but corrects out the estimated effects of `fs_version` (from mu term) and `study` (from mu & sigma terms) parameters.

Expects `fs_version` to be a fixed effect and `study` to be a random effect fit using `re()` function!!!

```
pred.cor <- centile_predict.corrected(sGMV.re, sim$dataToPredictM, sim$dataToPredictF, sim$ageRange, si
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

```
names(pred.cor)
```

```
## [1] "fanCentiles"    "fanCentiles_M"  "fanCentiles_F"  "peak"  
## [5] "peak_age"       "peak_M"        "peak_age_M"    "peak_F"  
## [9] "peak_age_F"     "M_mu"         "M_sigma"      "M_nu"  
## [13] "F_mu"          "F_sigma"      "F_nu"
```

`GGalt.variance` - copied from Simon's Nature paper repo

```
var <- GGalt.variance(pred$M_mu, pred$M_sigma, pred$M_nu)
```

Plotting

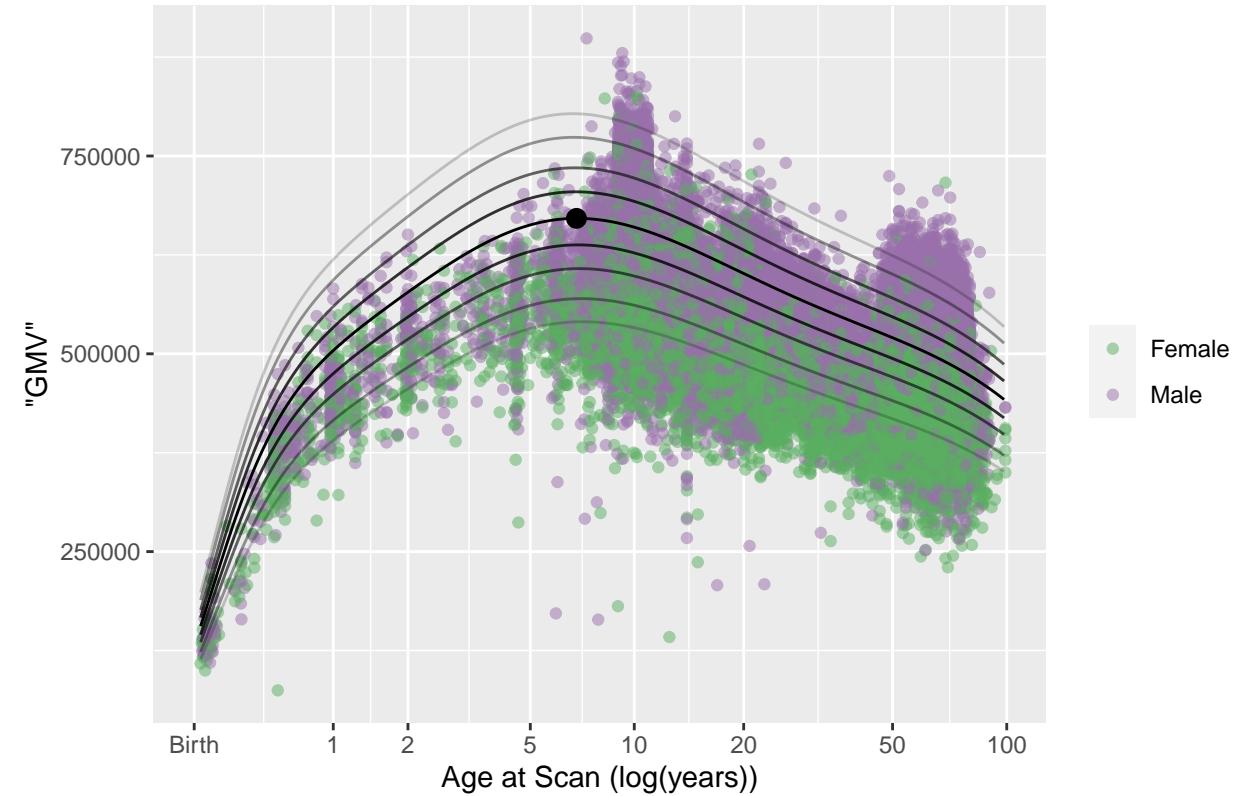
`makeCentileFan()` - basic centile fan plotting that averages across sex and predicts on `Mode(fs_version)` and `Mode(study)` of original data the gamlss was modeled on. Expects GAMLSS model, phenotype being modeled, and the name of the original df.

`age_transformed` parameter set to TRUE or FALSE

```
makeCentileFan(GMV.int, "GMV", cn_df, TRUE, "sex")
```

```
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
```

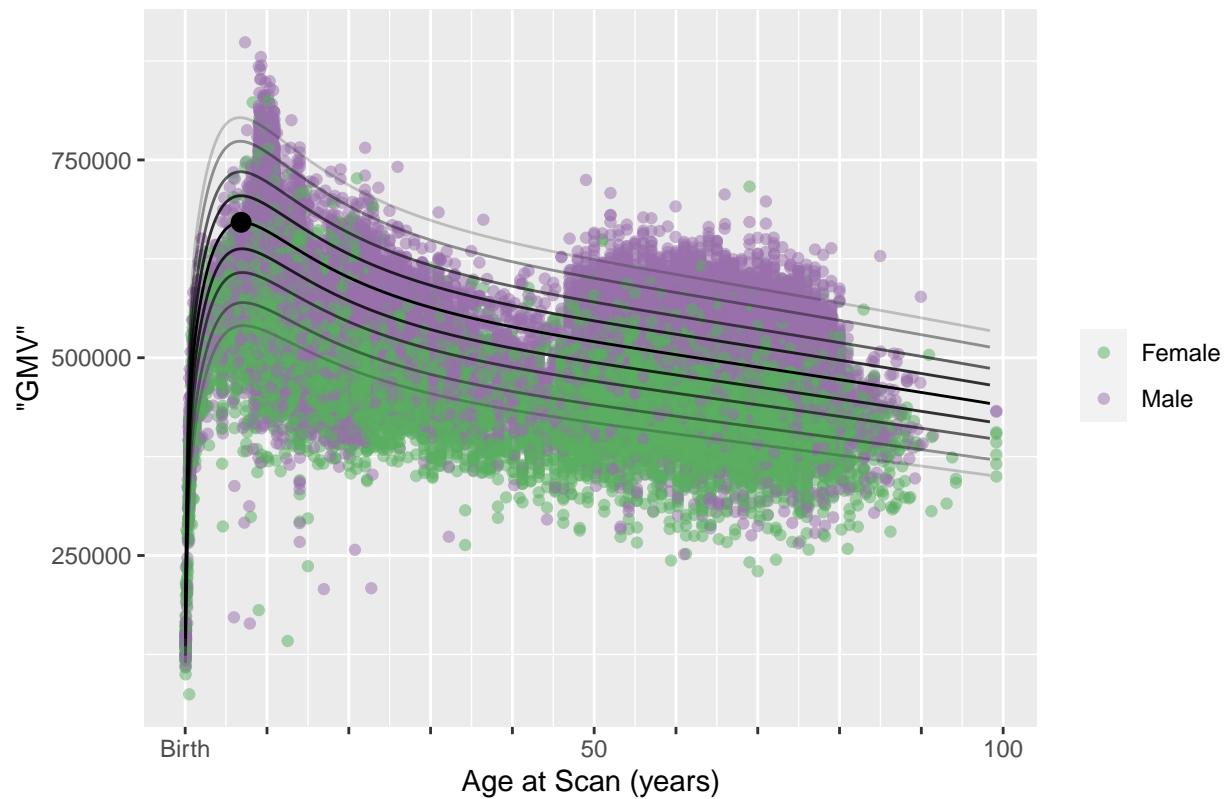
GMV.int



```
makeCentileFan(GMV.int, "GMV", cn_df, FALSE, "sex")
```

```
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
```

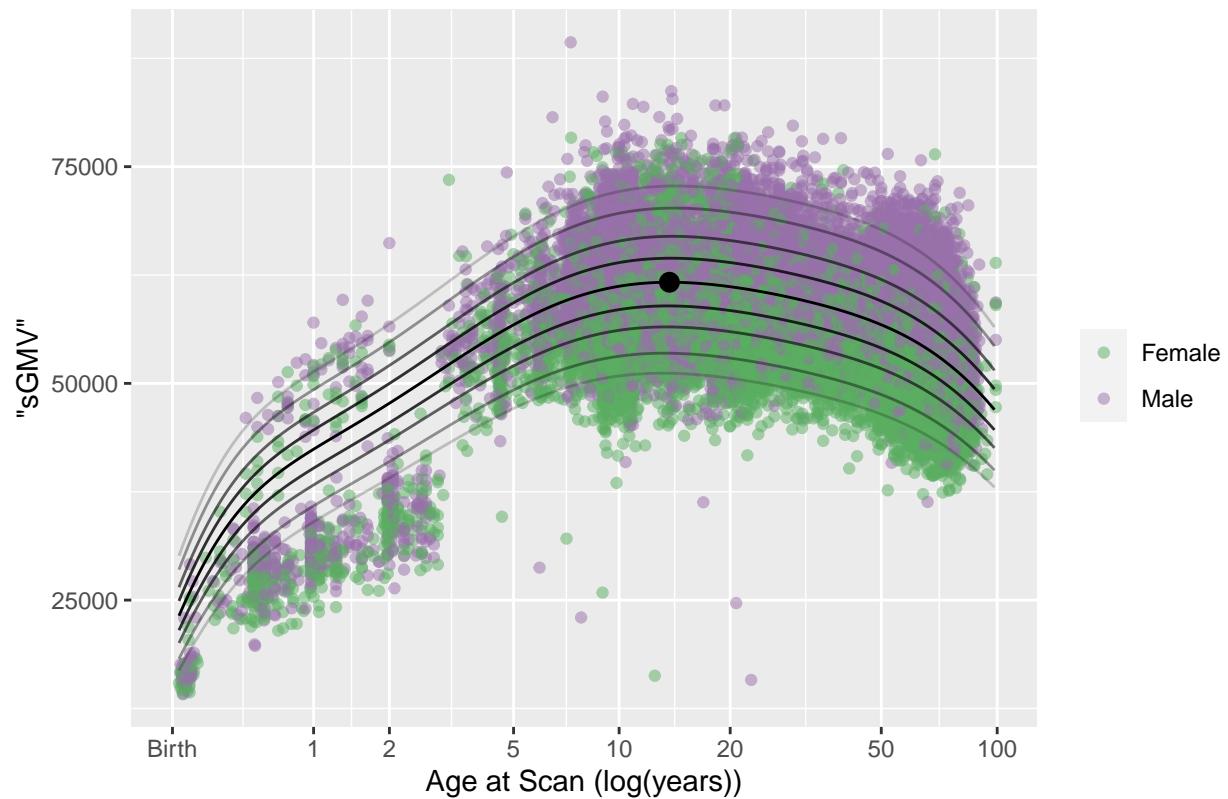
GMV.int



```
makeCentileFan(sGMV.re, "sGMV", cn_df, TRUE, "sex")
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

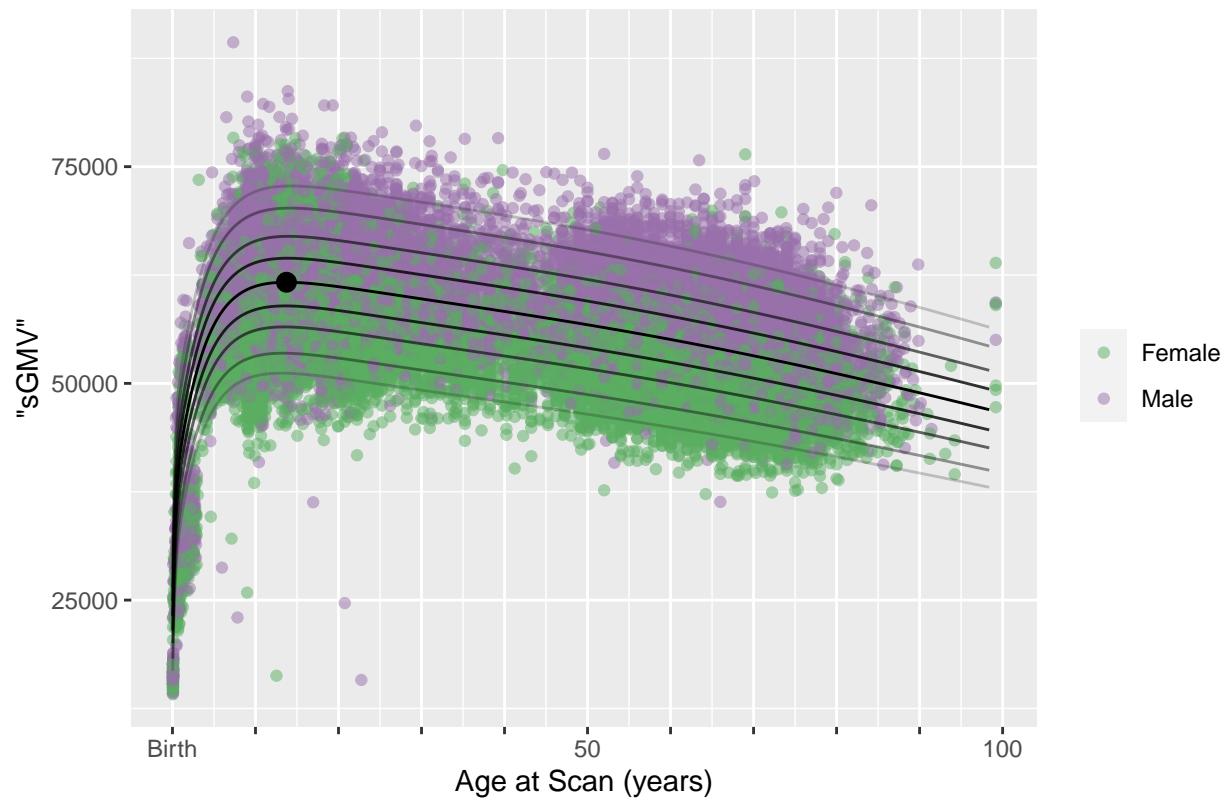
sGMV.re



```
makeCentileFan(sGMV.re, "sGMV", cn_df, FALSE, "sex")
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

sGMV.re

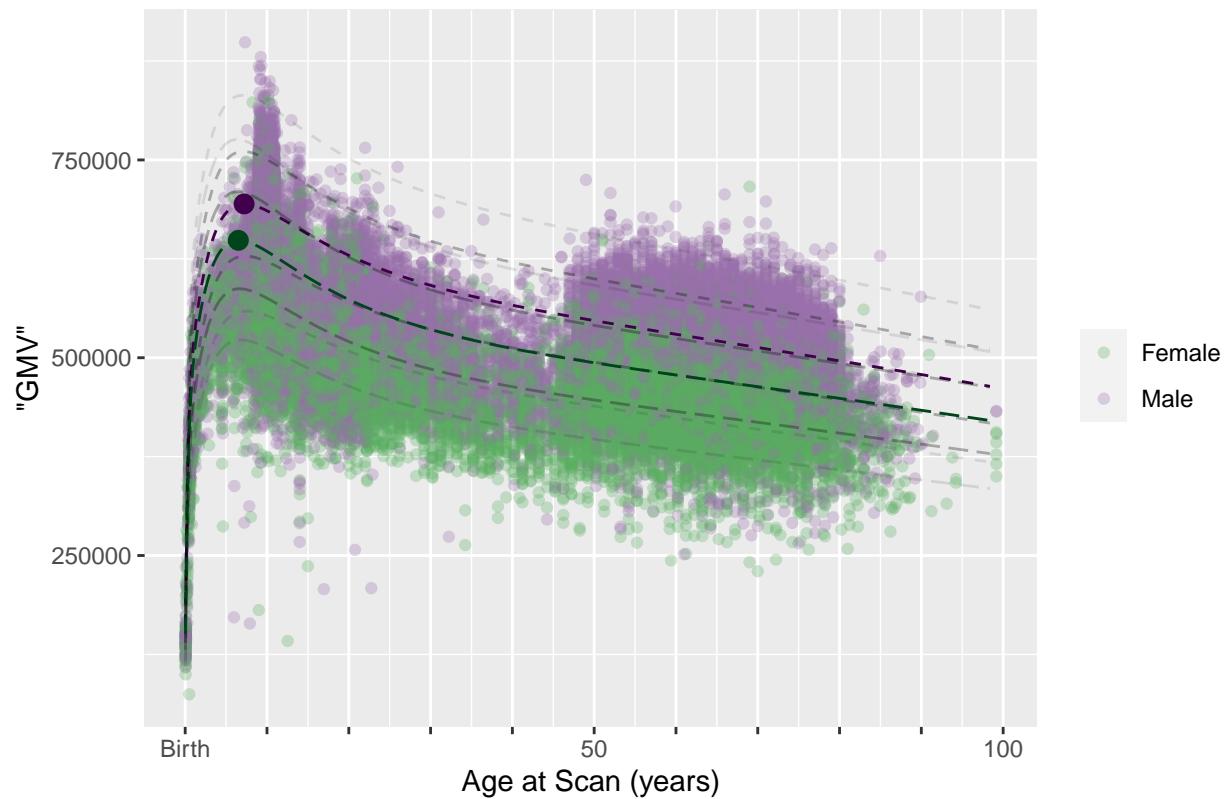


`makeCentileFan_sex_overlay()` - same as `makeCentileFan` but with separate centile lines for males and females

```
makeCentileFan_sex_overlay(GMV.int, "GMV", cn_df, FALSE, "sex")
```

```
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
```

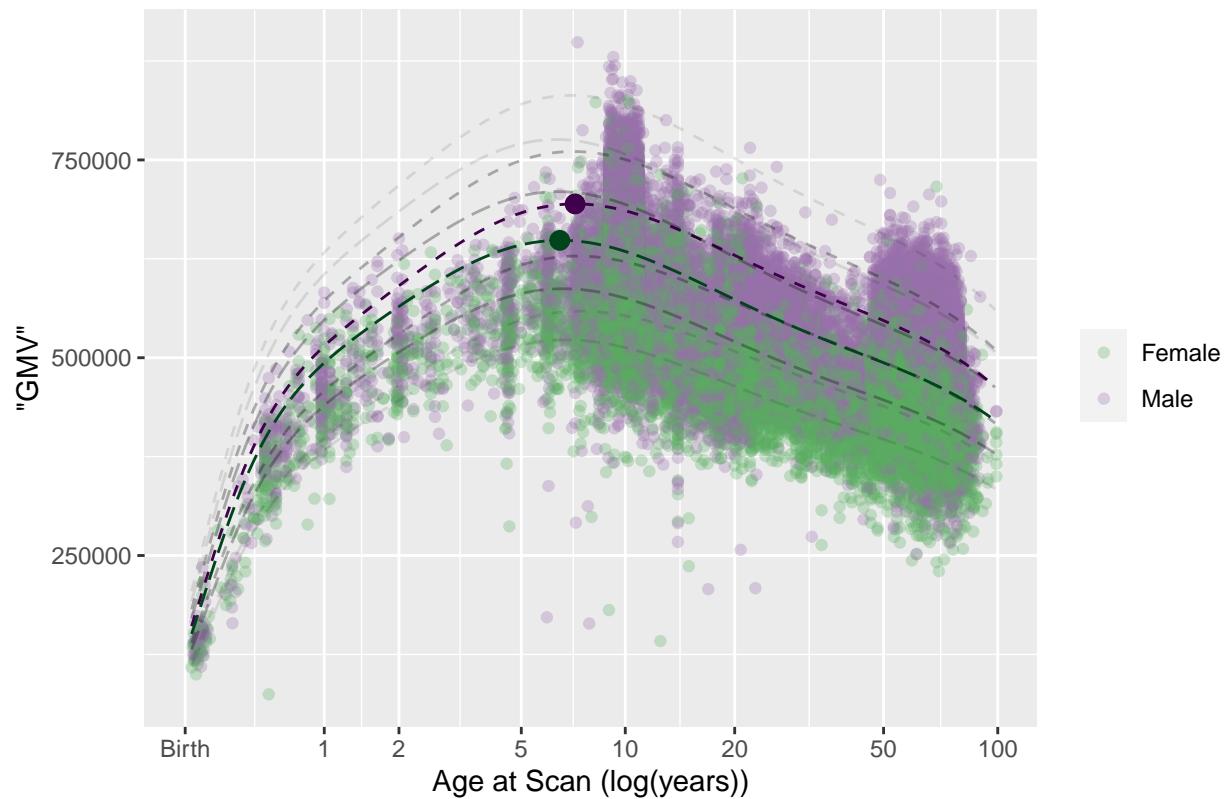
GMV.int



```
makeCentileFan_sex_overlay(GMV.int, "GMV", cn_df, TRUE, "sex")
```

```
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
```

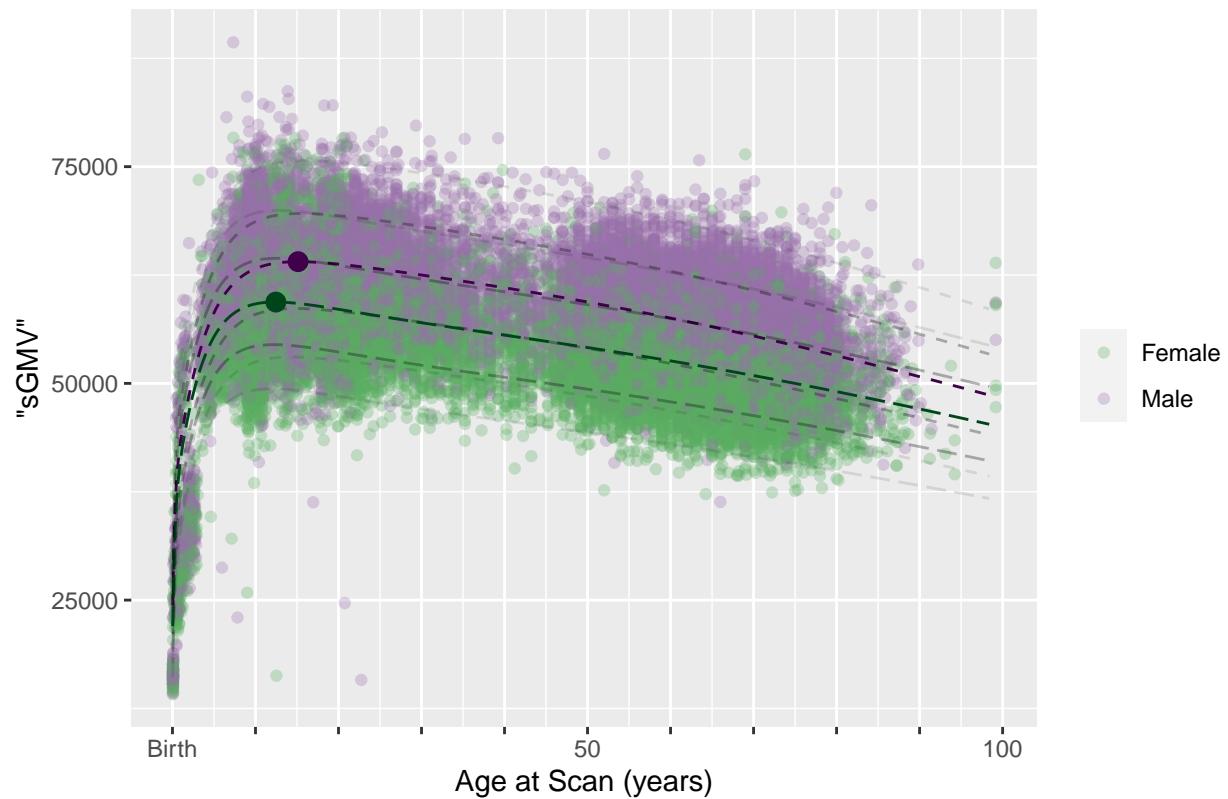
GMV.int



```
makeCentileFan_sex_overlay(sGMV.re, "sGMV", cn_df, FALSE, "sex")
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

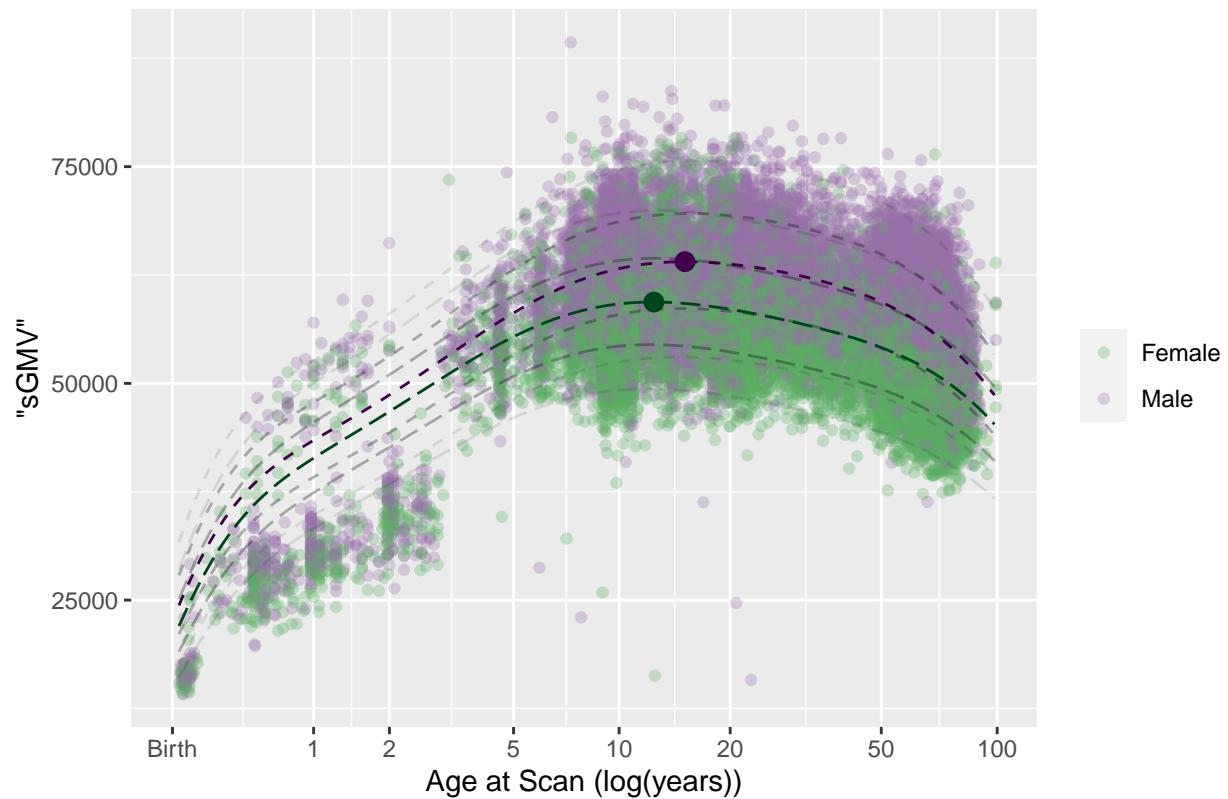
sGMV.re



```
makeCentileFan_sex_overlay(sGMV.re, "sGMV", cn_df, TRUE, "sex")
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

sGMV.re



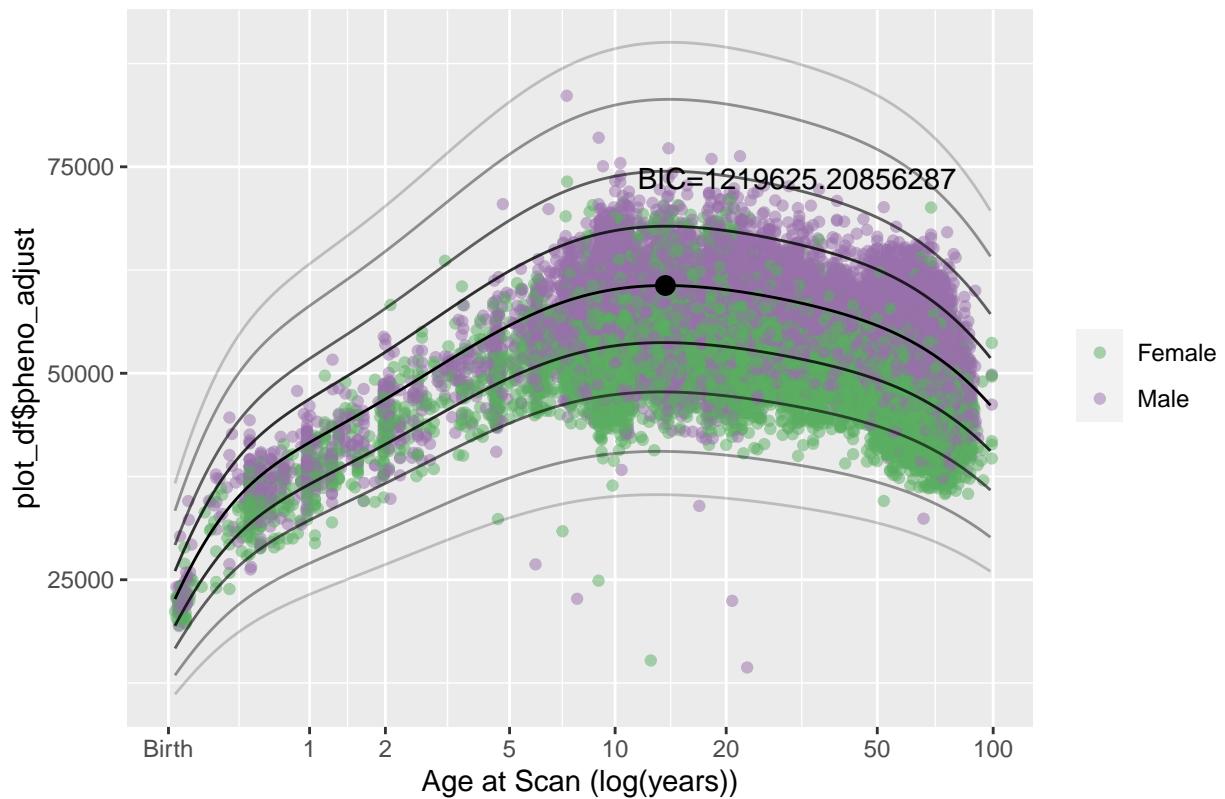
`makeCentileFan.corrected()` - centile fan plot that averages across sexes, correcting for `fs_version` and study effects in both centiles and data points plotted.

Requires `fs_version` to be a fixed effect and `study` to be a random effect fit using `re()` function!!!
age_transformed parameter set to TRUE or FALSE

```
makeCentileFan.corrected(sGMV.re, "sGMV", cn_df, TRUE, "sex")
```

```
## new prediction
## new prediction
## new prediction
## new prediction
```

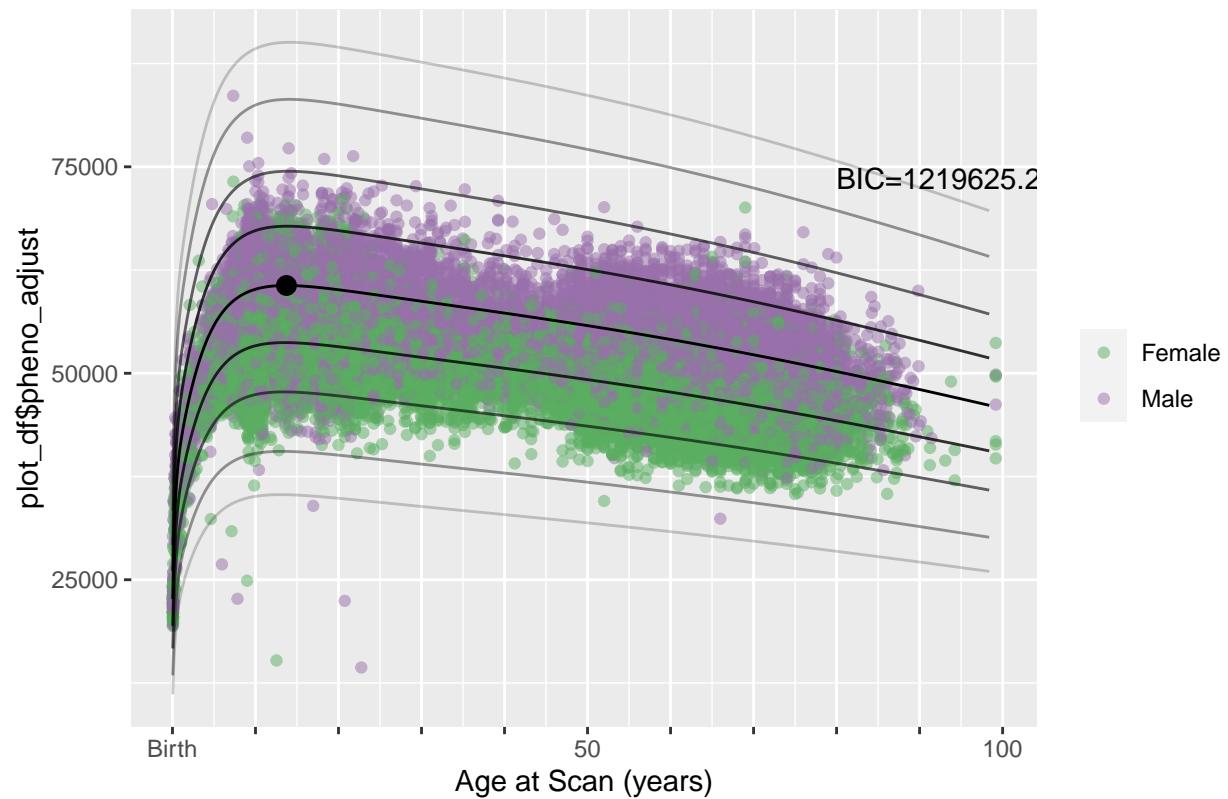
sGMV.re



```
#makeCentileFan(sGMV.re, "sGMV", cn_df, TRUE, "sex")
makeCentileFan.corrected(sGMV.re, "sGMV", cn_df, FALSE, "sex")
```

```
## new prediction
## new prediction
## new prediction
## new prediction
```

sGMV.re

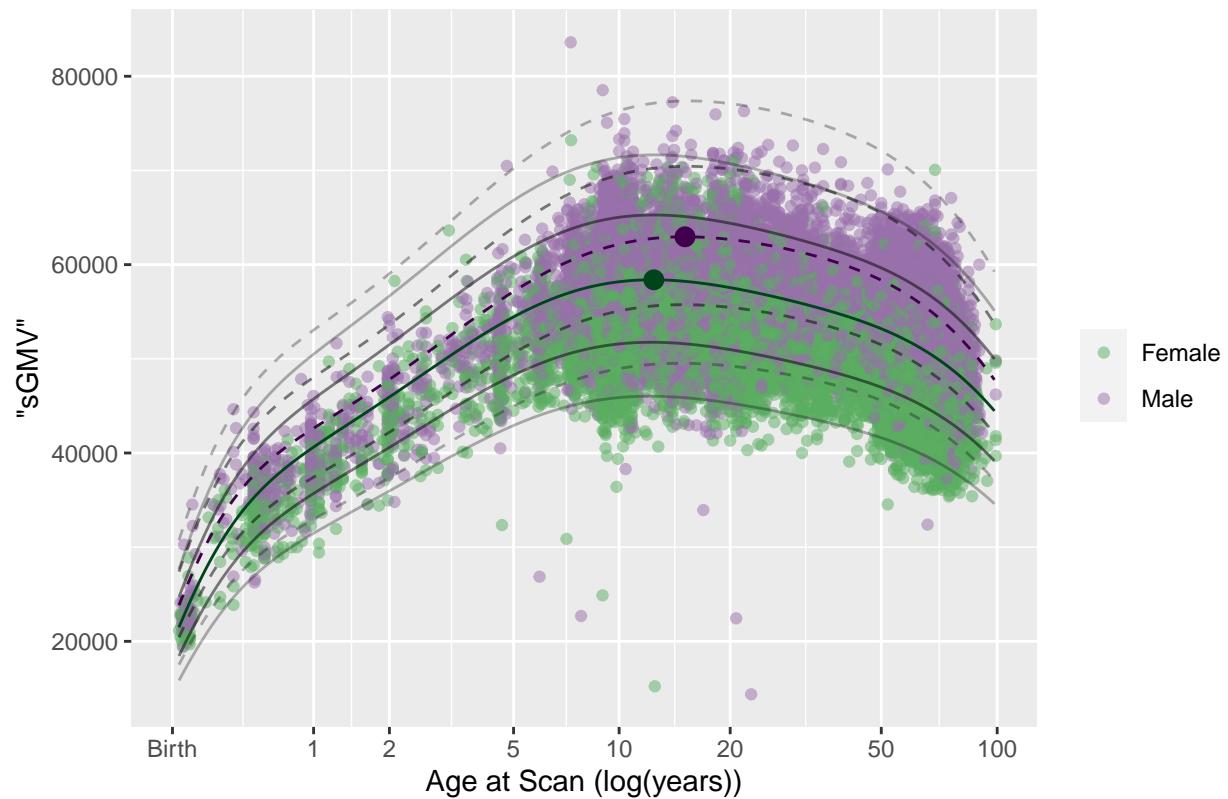


`makeCentileFan_sex_overlay.corrected()` - same as `makeCentileFan.corrected` but with separate centile lines for males and females

```
makeCentileFan_sex_overlay.corrected(sGMV.re, "sGMV", cn_df, TRUE, "sex")
```

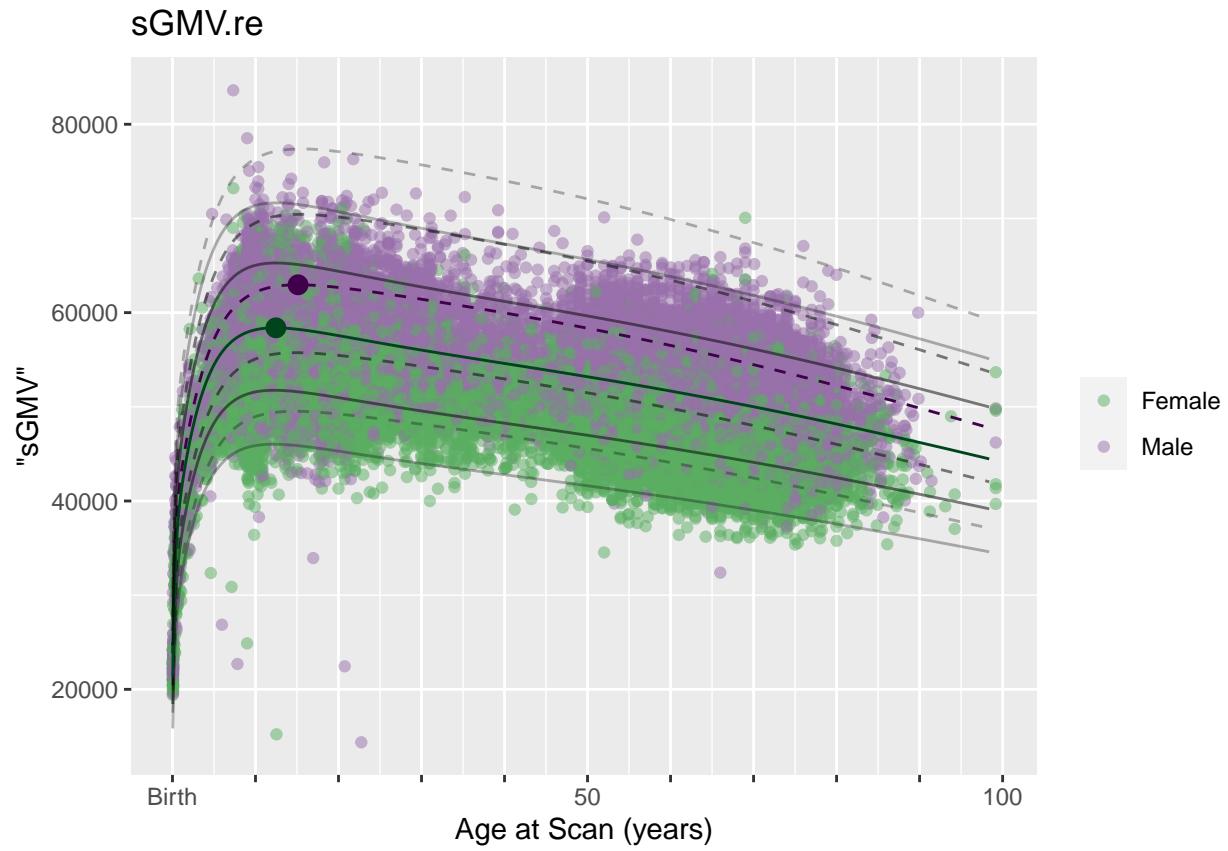
```
## new prediction
## new prediction
## new prediction
## new prediction
```

sGMV.re



```
makeCentileFan_sex_overlay.corrected(sGMV.re, "sGMV", cn_df, FALSE, "sex")
```

```
## new prediction  
## new prediction  
## new prediction  
## new prediction
```

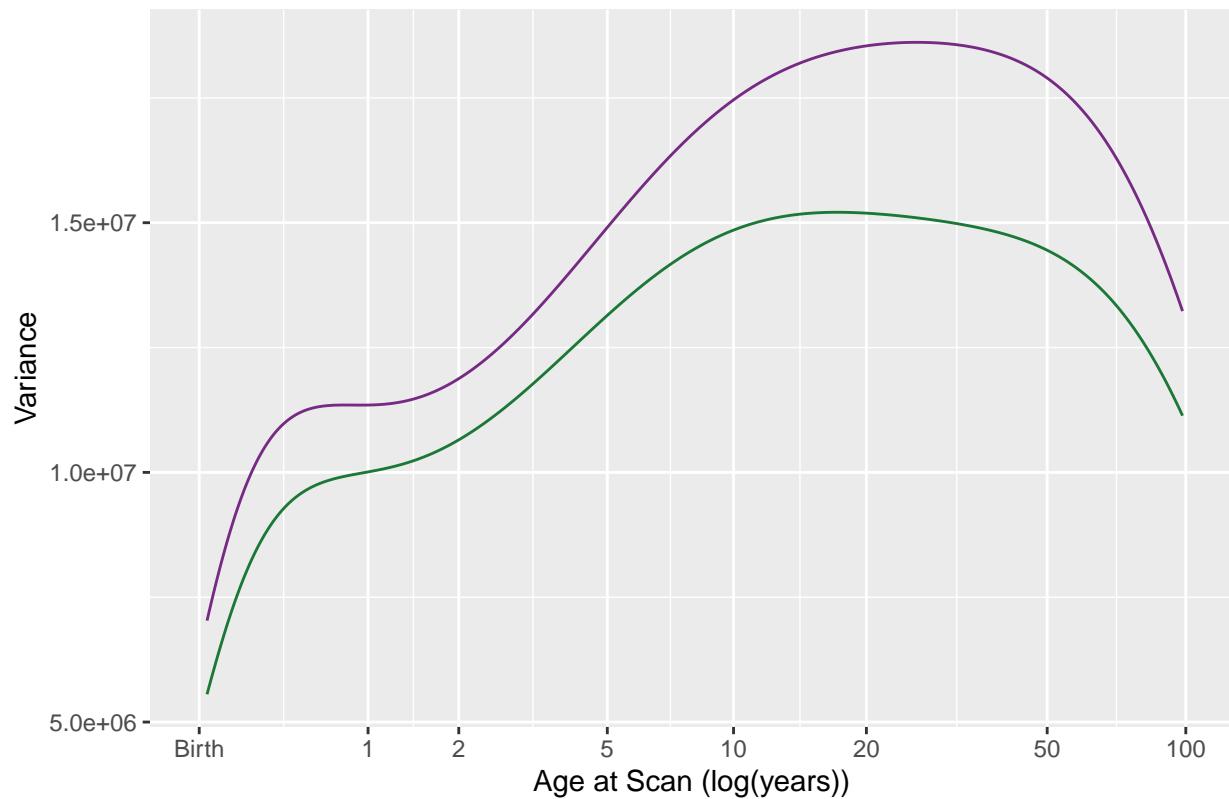


`plot.gamlss.var()` - plots variance from predicted GAMLSS model separately for males and females.
age_transformed parameter set to TRUE or FALSE

```
plot.gamlss.var(sGMV.re, "sGMV", cn_df, TRUE)
```

```
## new prediction
## new prediction
## new prediction
## new prediction
```

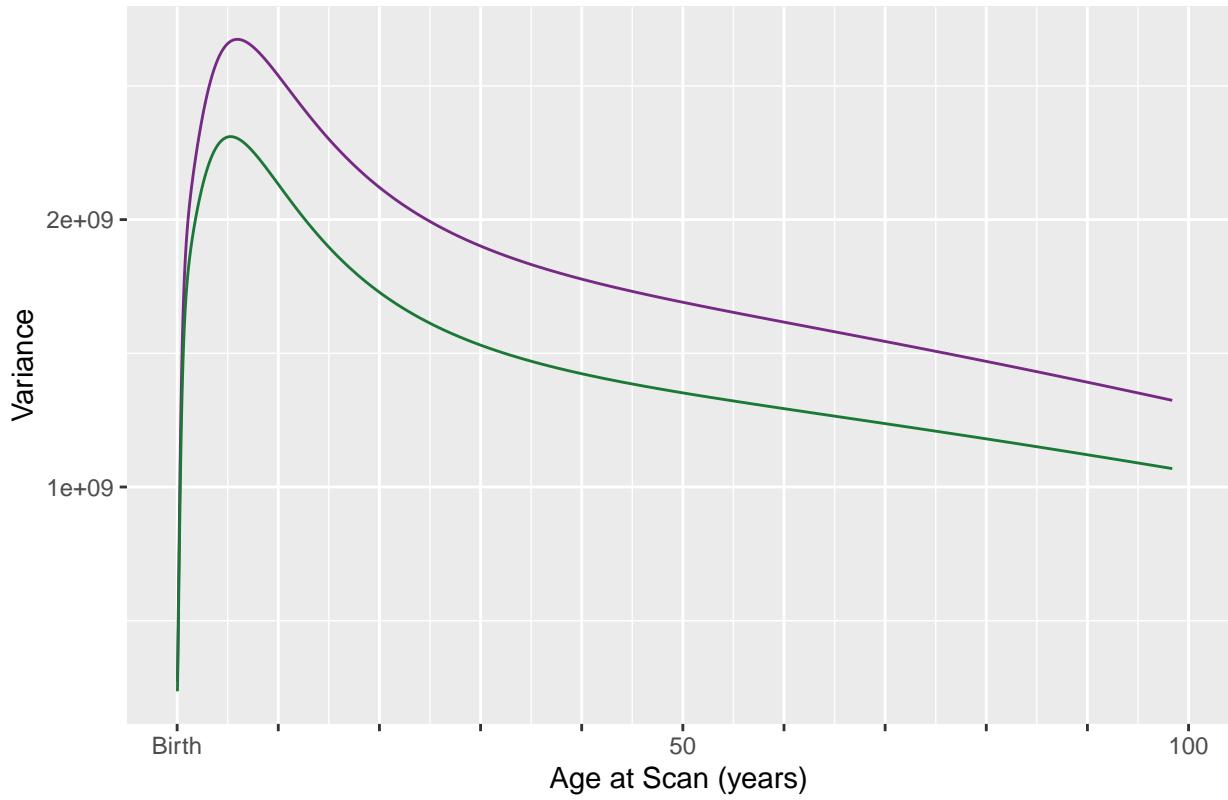
sGMV variance



```
plot.gamlss.var(GMV.int, "GMV", cn_df, FALSE)
```

```
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
```

GMV variance



`correct.points()` - function used within site/fs_version corrected plotting to account for these parameter's estimate's effects on each data point. Expects `re()` to be used when fitting random effect of site. Can also be called on it's own. Returns new df with each subject's phenotype corrected.

```
plot_df <- correct.points(sGMV.re, "GMV", cn_df)
summary(plot_df)
```

```
##      GMV          sGMV          WMV          Ventricles
##  Min.   : 74583   Min.   :14132   Min.   : 49733   Min.   :  811
##  1st Qu.:465826  1st Qu.:52392  1st Qu.:411303  1st Qu.:11296
##  Median :510770  Median :56277  Median :451791  Median :17351
##  Mean    :519429  Mean   :56374  Mean   :453583  Mean   :21542
##  3rd Qu.:565140  3rd Qu.:60497  3rd Qu.:495940  3rd Qu.:27358
##  Max.    :898654  Max.   :89344  Max.   :1064862 Max.   :175416
##
##      session       participant       run        study
##  Min.   : 58764   Min.   : 1       Min.   :1.000  Length:62447
##  1st Qu.: 1766    1st Qu.:21374  1st Qu.:1.000  Class  :character
##  Median : 1013    Median :41051  Median :1.000  Mode   :character
##  Mean   : 111     Mean   :40829  Mean   :1.025
##  3rd Qu.:  89     3rd Qu.:61084  3rd Qu.:1.000
##  Max.   :  73     Max.   :80991  Max.   :8.000
##  (Other): 631
##
##      fs_version      age_days      sex        dx        country
##  Custom  : 674     Min.   : 291  Female:31646  CN:62447   UK      :30081
##  FS53    : 3571    1st Qu.: 5393  Male  :30801           USA     :20734
```

```

##  FS6_T1  :18219  Median :19273          France : 1816
##  FS6_T1T2:39335  Mean   :15698          China  : 1268
##  FSIInfant: 648  3rd Qu.:24321          India   : 1193
##                               Max.   :36504          Germany: 1036
##                                         (Other): 6319
##
##      site       age_yrs      log_age       TBV
##  11025    :17617  Min.   : 0.7967  Min.   :2.464  Min.   : 136632
##  11027    : 7429  1st Qu.:14.7664  1st Qu.:3.732  1st Qu.: 917810
##  11026    : 3985  Median :52.7659  Median :4.285  Median : 993084
##  1        : 2469  Mean   :42.9796  Mean   :4.075  Mean   : 994554
##  MRI-Share: 1341  3rd Qu.:66.5878  3rd Qu.:4.386  3rd Qu.:1072189
##  PNC      : 1217  Max.   :99.9425  Max.   :4.562  Max.   :1706546
##  (Other)  :28389
##
##      log_TBV      log_GMV      log_sGMV      log_WMV
##  Min.   :5.136  Min.   :4.873  Min.   :4.150  Min.   :4.697
##  1st Qu.:5.963  1st Qu.:5.668  1st Qu.:4.719  1st Qu.:5.614
##  Median :5.997  Median :5.708  Median :4.750  Median :5.655
##  Mean   :5.994  Mean   :5.710  Mean   :4.748  Mean   :5.651
##  3rd Qu.:6.030  3rd Qu.:5.752  3rd Qu.:4.782  3rd Qu.:5.695
##  Max.   :6.232  Max.   :5.954  Max.   :4.951  Max.   :6.027
##
##      log_Ventricles     site_int     pheno_adjust     fs_effect
##  Min.   :2.909  Min.   :-0.315826  Min.   : 60252  Min.   :-0.26998
##  1st Qu.:4.053  1st Qu.:-0.009192  1st Qu.:428767  1st Qu.: 0.07411
##  Median :4.239  Median : 0.008215  Median :470435  Median : 0.07865
##  Mean   :4.250  Mean   : 0.001359  Mean   :481721  Mean   : 0.07444
##  3rd Qu.:4.437  3rd Qu.: 0.008215  3rd Qu.:523826  3rd Qu.: 0.07865
##  Max.   :5.244  Max.   : 0.213378  Max.   :840912  Max.   : 0.10626
##

```

Experimental

Log-Log(TBV) correction

This function takes $\log(\text{phenotype}) \sim \log(\text{TBV})$ models and plots $\log(\text{pheno})$ by age. It predicts on simulated data where for each age, \log_{TBV} is the mean \log_{TBV} (or roughly imputed mean) for that sex at that given age. Hence the “experimental” label, since, as you can see, it’s pretty rough.

```

## GAMLSS-RS iteration 1: Global Deviance = -324689.2
## GAMLSS-RS iteration 2: Global Deviance = -325130.6
## GAMLSS-RS iteration 3: Global Deviance = -325139.9
## GAMLSS-RS iteration 4: Global Deviance = -325140.3
## GAMLSS-RS iteration 5: Global Deviance = -325140.5
## GAMLSS-RS iteration 6: Global Deviance = -325140.6
## GAMLSS-RS iteration 7: Global Deviance = -325140.7
## GAMLSS-RS iteration 8: Global Deviance = -325140.7
## GAMLSS-RS iteration 9: Global Deviance = -325140.7
## GAMLSS-RS iteration 10: Global Deviance = -325140.8
## GAMLSS-RS iteration 11: Global Deviance = -325140.8
## GAMLSS-RS iteration 12: Global Deviance = -325140.8
## GAMLSS-RS iteration 13: Global Deviance = -325140.8
## GAMLSS-RS iteration 14: Global Deviance = -325140.8
## GAMLSS-RS iteration 15: Global Deviance = -325140.8

```



```
makeCentileFan_sex_overlay.logTBV(log_wmv_refit, "log_WMV", cn_df, TRUE, "sex")
```

```

## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)
## new prediction
## New way of prediction in random() (starting from GAMLSS version 5.0-6)

```

