06_update

randy

2022-04-19

1. Redo table 1 with final sample

```
table0
```

Variable	**N = 1,370**[1]
Genotype	
One allele F508del	477 (35%)
Others or Unknown	150 (11%)
Two alleles F508del	743 (54%)
Gender	
Female	693 (51%)
Male	677 (49%)
Race	
Other	62 (4.5%)
White	1,308 (95%)
Ethnicity	
Hispanic	43 (3.3%)
Non-Hispanic	1,274 (97%)
Unknown	53
Visit number	45 (32, 60)
Age at registration, years	3.13 (3.06, 3.22)
Age at end of follow-up, years	12.8 (10.2, 16.0)
Follow-up, years	9.6 (7.0, 12.7)
Height at registration, cm	94.0 (91.4, 97.0)
Weight at registration, kg	14.10 (13.10, 15.30)

[1] n (%); Median (IQR)

```
table1
```

Variable	**testing, N = 457**[1]	**training, N = 913**[1]
Genotype		
One allele F508del	151 (33%)	326 (36%)
Others or Unknown	58 (13%)	92 (10%)
Two alleles F508del	248 (54%)	495 (54%)
Gender		
Female	224 (49%)	469 (51%)
Male	233 (51%)	444 (49%)
Race		
Other	21 (4.6%)	41 (4.5%)
White	436 (95%)	872 (96%)
Ethnicity		
Hispanic	18 (4.1%)	25 (2.8%)
Non-Hispanic	419 (96%)	855 (97%)
Unknown	20	33
Visit number	44 (32, 61)	45 (31, 59)
Age at registration, years	3.14 (3.07, 3.23)	3.13 (3.05, 3.22)
Age at end of follow-up, years	13.1 (10.3, 15.9)	12.7 (10.2, 16.0)
Follow-up, years	9.9 (7.2, 12.7)	9.4 (7.0, 12.9)
Height at registration, cm	94.0 (91.7, 97.2)	94.0 (91.2, 96.8)
Weight at registration, kg	14.10 (13.20, 15.30)	14.10 (13.10, 15.30)

[1] n (%); Median (IQR)

```
## save pptx -----------------------------------------------
## flextable can be saved directly to powerpoints
flextable::save_as_pptx(
  table0,
  path = "figure/01_table0.pptx")
flextable::save_as_pptx(
  table1,
  path = "figure/01_table1.pptx")
```

2. Refine spline model for training dataset.

Try spline model with 3 inner knots set at percentiles (quartiles) too and compare to the splines you had tried before.

```r
train <- here::here("data",
                    "epic_clean_randy.csv") %>%
  read.csv(row.names = 1) %>%
  mutate(id = as.factor(id)) %>%
  filter(group == "training")

test <- here::here("data",
                   "epic_clean_randy.csv") %>%
  read.csv(row.names = 1) %>%
  filter(group == "testing")

basetest <- test %>%
  group_by(id) %>%
  arrange(time) %>%
  slice(1L)
```

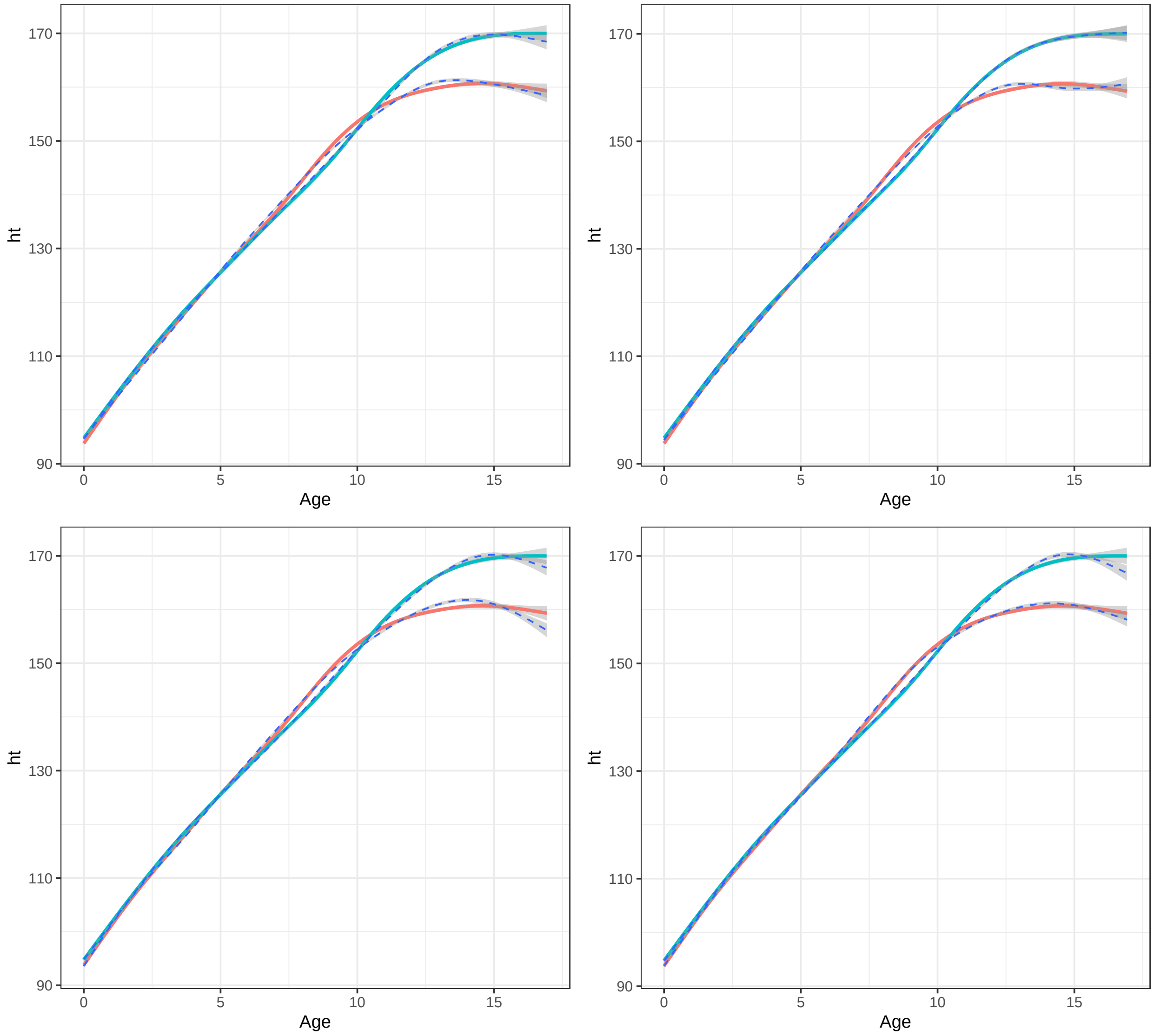
```r
# lmeControl
ctrl <- lmeControl(opt = 'optim')

## the original mod2_2_4 in file "04_model_fit.Rmd"
fit2_0 <- lme(ht ~ bs(time, knots = c(10, 15),
                      degree = 3,
                      intercept = FALSE) * sex +  genotype - 1,
              control = ctrl,
              random = ~ 1| id,
              data = train)

fit2_1 <- lme(ht ~ bs(time, knots = c(10, 12),
                      degree = 3,
                      intercept = FALSE) * sex +  genotype - 1,
              control = ctrl,
              random = ~ 1| id,
              data = train)

fit2_2 <- lme(ht ~ bs(time, df = 5,
                      degree = 3,
                      intercept = FALSE) * sex +  genotype - 1,
              control = ctrl,
              random = ~ 1| id,
              data = train)

fit3_0 <- lme(ht ~ bs(time, df = 6,
                      degree = 3,
                      intercept = FALSE) * sex +  genotype - 1,
              control = ctrl,
              random = ~ 1| id,
              data = train)
```

```r
get_tag <- function(fit, data1 = train) {
  tidy <- broom.mixed::tidy(fit)
  glance <- broom.mixed::glance(fit)
  augment <- broom.mixed::augment(fit) %>%
    dplyr::select(time, .fitted, sex) %>%
    unique()

  plot1 <- ggplot() +
        geom_smooth(data = data1,
                    aes(x = time,
                        y = ht,
                        group = sex,
                        color = sex)) +
        geom_smooth(data = augment,
                    aes(x = time, y = .fitted,
                        group = sex),
                    linetype = 2,
                    size = 0.5) +
        xlab("Age") +
        theme_bw() +
    theme(legend.position = "none")


  return(list(tidy = tidy,
              glance = glance,
              augment = augment,
              plot = plot1))
}
```

```
tag2_0 <- get_tag(fit = fit2_0)
tag2_1 <- get_tag(fit = fit2_1)
tag2_2 <- get_tag(fit = fit2_2)
tag3_0 <- get_tag(fit = fit3_0)

glance_all <- map_dfr(list(tag2_0, tag2_1,
                    tag2_2, tag3_0),
                  "glance") %>%
  mutate(model = c("fit2_0", "fit2_1",
                   "fit2_2", "fit3_0")) %>%
  dplyr::select(model, everything())

glance_all
```

```
## # A tibble: 4 x 6
##   model   nobs sigma   logLik     AIC     BIC
##   <chr>  <int> <dbl>    <dbl>   <dbl>   <dbl>
## 1 fit2_0 43645  2.52 -104694. 209421. 209560.
## 2 fit2_1 43645  2.49 -104297. 208626. 208765.
## 3 fit2_2 43645  2.50 -104488. 209008. 209147.
## 4 fit3_0 43645  2.49 -104178. 208393. 208549.
```

3. plot the predicted/expected means.



4. For brokenstick model try time for a anchoring of 10 and 12

```
# View(brokenstick)
# View(brokenstick::brokenstick_bridge)

bks1 <- brokenstick(formula = ht ~ time | id,
                    data = train,
                    knots = c(10, 15),
                    method = "lmer")

bks10 <- brokenstick(formula = ht ~ time | id,
                    data = train,
                    subjid = train$id,
                    knots = c(10, 15))

est1_knots <- predict(bks1,
                     x = "knots",
                     group = train$id)
est1_all <- predict(bks1,
                   group = train$id)


bks2 <- brokenstick(formula = ht ~ time | id,
                    data = train,
                    subjid = train$id,
                    knots = c(10, 12))
est2_knots <- predict(bks2,
                     x = "knots",
                     group = train$id)
est2_all <- predict(bks2,
                   group = train$id)


bks31 <- brokenstick(formula = ht ~ time | id,
                    data = train,
                    subjid = train$id,
                    knots = c(10, 12, 15))

bks32 <- brokenstick(formula = ht ~ time | id,
                    data = train,
                    knots = c(10, 12, 15),
                    seed = 555)

est31_knots <- predict(bks31,
                     x = "knots",
                     group = train$id)

est31_all <- predict(bks31,
                   group = train$id)

est32_knots <- predict(bks32,
                     x = "knots",
                     group = train$id)
est32_all <- predict(bks32,
                   group = train$id)
```

the biometrics paper,

```
summary(bks31)
```

```
## Class        brokenstick (kr)
## Variables    ht (outcome), time (predictor), id (group)
## Data         43645 (n), 0 (nmis), 913 (groups)
## Parameters   22 (total), 5 (fixed), 5 (variance), 10 (covariance), 2 (error)
## Knots        0 10 12 15 16.93
## Means        95.72131 155.6023 163.3207 165.4326 164.6083
## Residuals    0.2421744 0.8802273 1.367782 2.204885 31.10522 (min, P25, P50, P75, max)
## Mean resid   1.946913
## R-squared    0.9957732
##
```

```
## Variance-covariance matrix
##                time_0    time_10    time_12    time_15 time_16.93
## time_0     19.5644154
## time_10    24.1895763 82.2681124
## time_12    17.4552221 50.5396343 81.4547876
## time_15    18.3407855 22.5970526 65.6704319 92.2207864
## time_16.93  15.534241 26.8396928 65.9984442 82.1386039 80.9523221
```

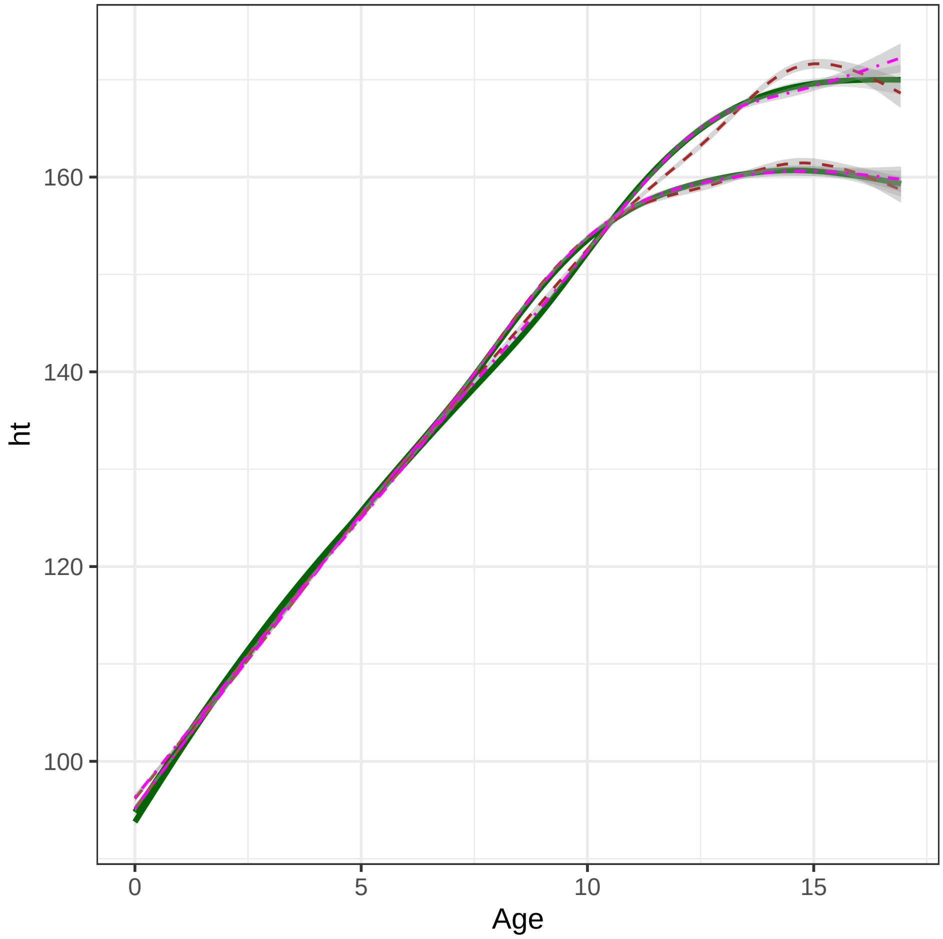
```
summary(bks32)
```

```
## Class        brokenstick (kr)
## Variables    ht (outcome), time (predictor), id (group)
## Data         43645 (n), 0 (nmis), 913 (groups)
## Parameters   22 (total), 5 (fixed), 5 (variance), 10 (covariance), 2 (error)
## Knots        0 10 12 15 16.93
## Means        95.71299 155.5664 163.2632 165.3464 164.0733
## Residuals    0.2440794 0.878987 1.387723 2.208051 30.83667 (min, P25, P50, P75, max)
## Mean resid   1.948516
## R-squared    0.9957725
##
## Variance-covariance matrix
##                time_0    time_10    time_12    time_15 time_16.93
## time_0     19.5935051
## time_10    24.2353492 82.6316341
## time_12    17.8177299 50.6881557  81.118866
## time_15    18.9074014  23.210069 65.3710932 91.3853125
## time_16.93 16.5900516 22.8482092 61.3250568 83.4066696 81.9589513
```

the estimated c(10, 12) and c(10, 15)

```
plot_bks1 <- ggplot() +
  geom_smooth(data = train,
              aes(x = time, y = ht,
                  group = sex),
              alpha = 0.3,
              size = 1,
              color = "darkgreen") +
  geom_smooth(data = est1_all,
              aes(x = time, y = .pred,
                  group = sex),
              color = "brown",
              linetype = 2,
              size = 0.5) +
  geom_smooth(data = est2_all,
                    aes(x = time, y = .pred,
                        group = sex),
              color = "magenta",
              linetype = 4,
              size = 0.5) +
  xlab("Age") +
  theme_bw()
```

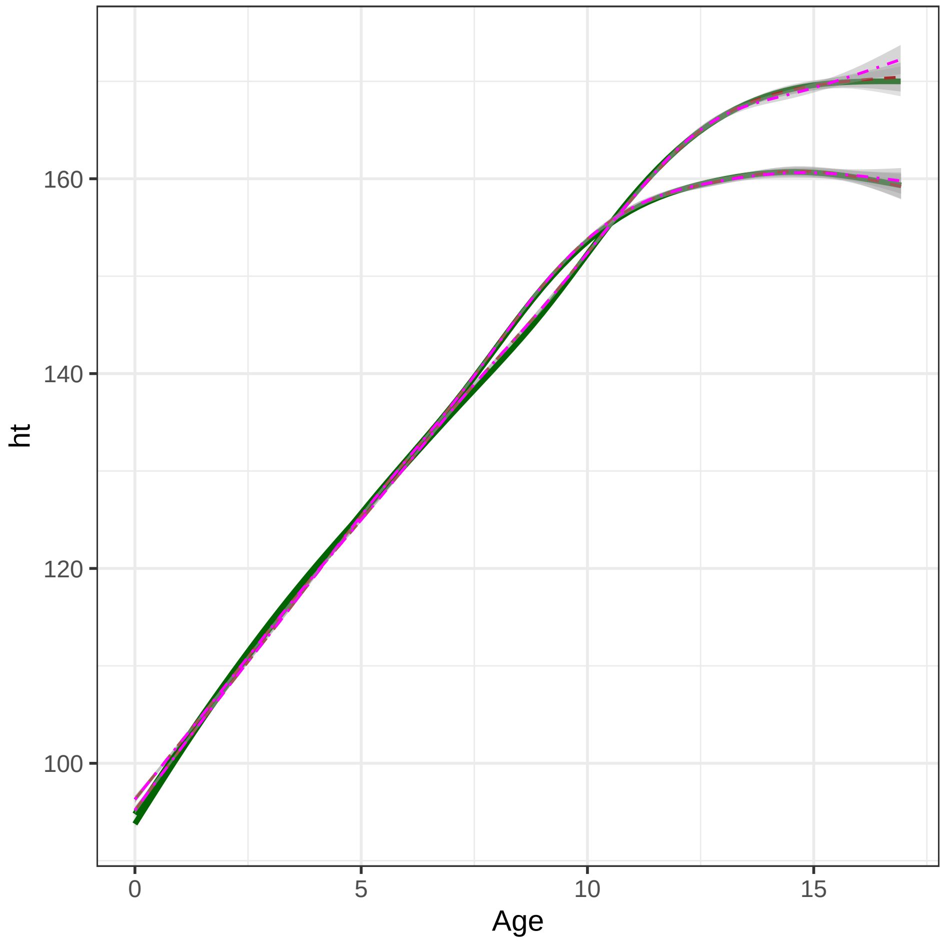
```
plot_bks1
```



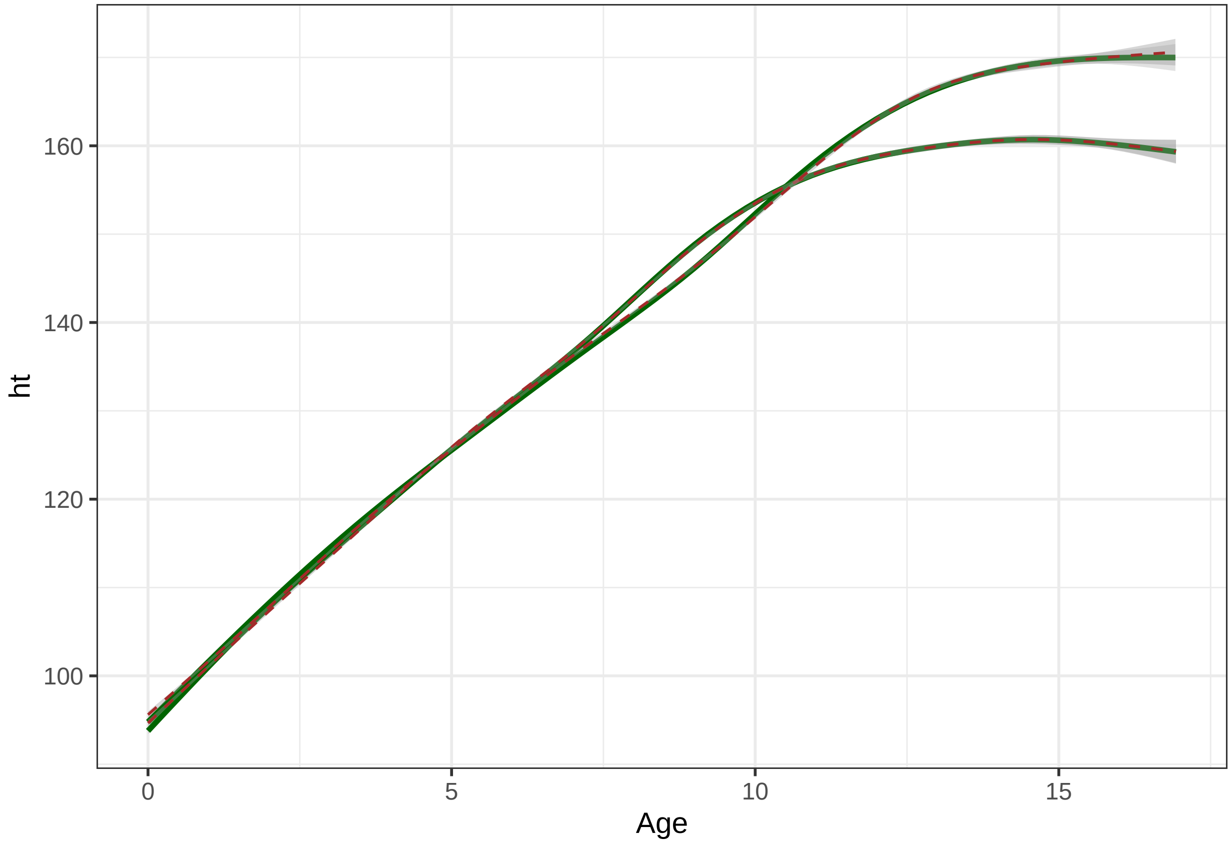
The estimated for 2 **c(10, 12)** and 3 c(10, 12, 15) knots

try c(10, 13), c(10, 14)

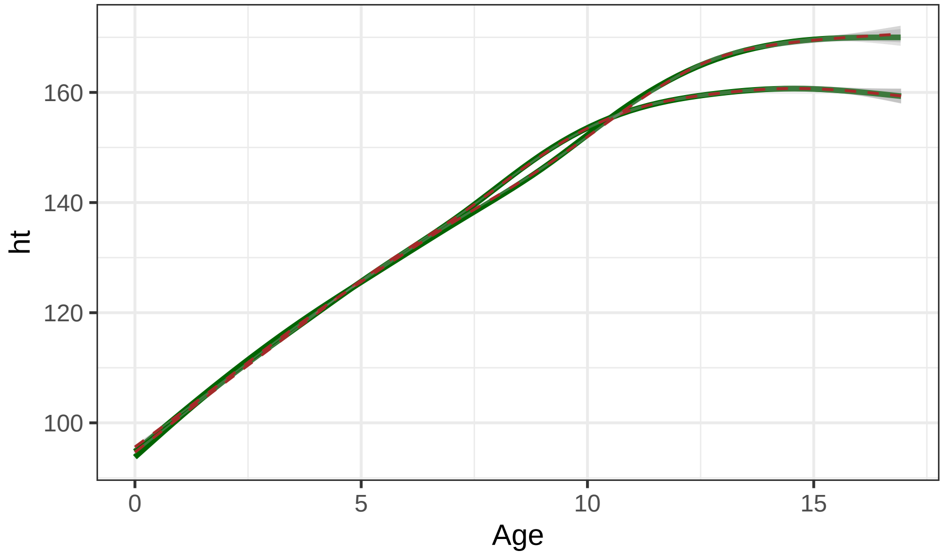
```
plot_bks2 <- ggplot() +
  geom_smooth(data = train,
              aes(x = time, y = ht,
                  group = sex),
              alpha = 0.3,
              size = 1,
              color = "darkgreen") +
  geom_smooth(data = est32_all,
              aes(x = time, y = .pred,
                  group = sex),
              color = "brown",
              linetype = 2,
              size = 0.5) +
  geom_smooth(data = est2_all,
                    aes(x = time, y = .pred,
                        group = sex),
              color = "magenta",
              linetype = 4,
              size = 0.5) +
  xlab("Age") +
  theme_bw()

plot_bks2
```



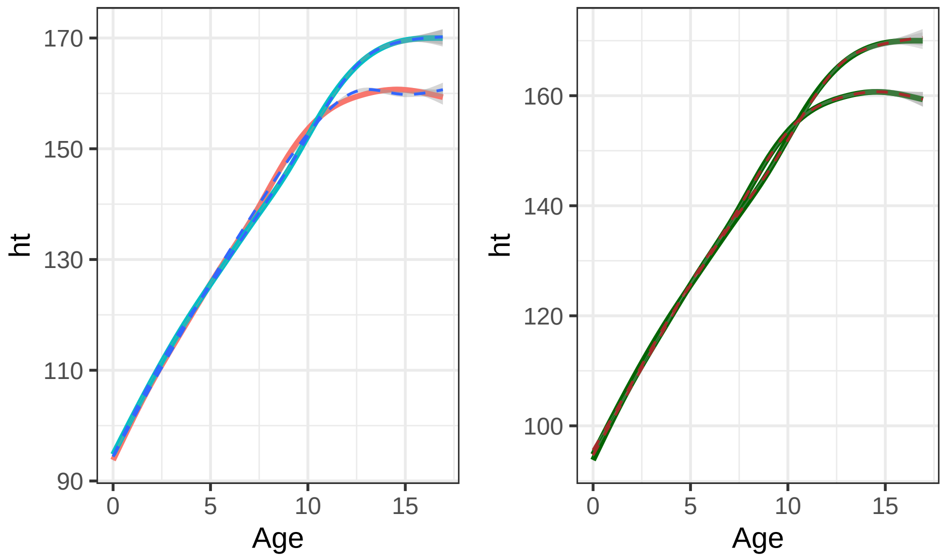
```
Kk <- c(5, 10, 12, 15)

Kplot <- function(K,
                  seed = 555,
                  form = "ht ~ time | id") {

  bks3n <- brokenstick(formula = as.formula(form),
                    data = train,
                    knots = Kk,
                    method = "lmer")
  # ?brokenstick
  est_knots <- predict(bks3n,
                     x = "knots",
                     group = train$id)
  est_all <- predict(bks3n,
                   group = train$id)

  plot_bks <- ggplot() +
    geom_smooth(data = train,
                aes(x = time, y = ht,
                    group = sex),
                alpha = 0.3,
                size = 1,
                color = "darkgreen") +
    geom_smooth(data = est_all,
                aes(x = time, y = .pred,
                    group = sex),
                color = "brown",
                linetype = 2,
                size = 0.5) +
    xlab("Age") +
    theme_bw()

  return(list(model = bks3n,
              est.knot = est_knots,
              est.all = est_all,
              plot = plot_bks))
}

bksn <- Kplot(K = Kk)
bksn$plot
```



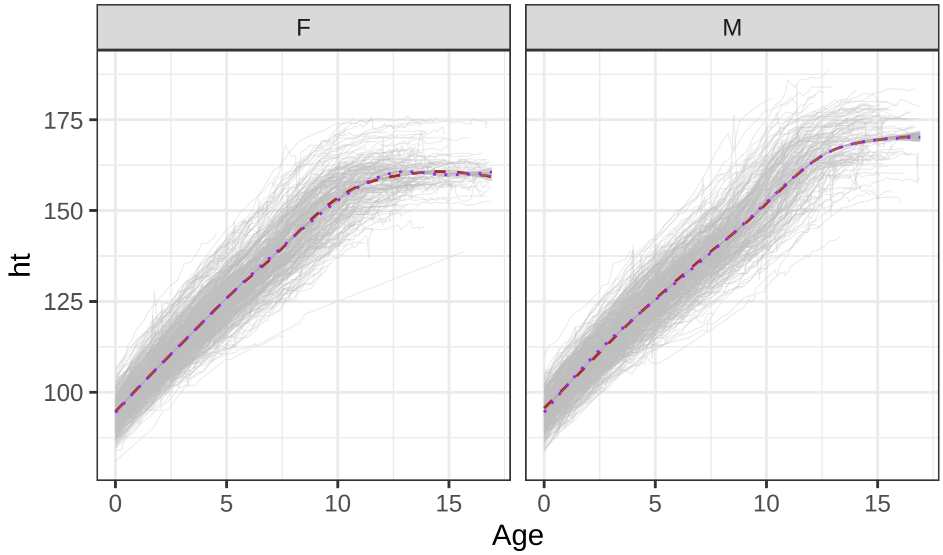
```
tag2_1 <- get_tag(fit = fit2_1)

bks2_1 <- Kplot(K = c(10, 12))
bks2_1$plot
```



```r
grid.arrange(tag2_1$plot,
             bks2_1$plot,
             ncol = 2)
```



```r
plot <- ggplot() +
  geom_line(data = train,
            aes(x = time,
                y = ht,
                group = id),
          color = "grey",
            alpha = 0.3,
            size = 0.2) +
  geom_smooth(data = bks2_1$est.all,
              aes(x = time, y = .pred),
              color = "brown",
              linetype = 2,
              size = 0.5) +
        geom_smooth(data = tag2_1$augment,
                  aes(x = time, y = .fitted,
                      group = sex),
                  linetype = 3,
                  color = "purple",
                  size = 0.5) +
  xlab("Age") +
  theme_bw()

plot + facet_wrap("sex")
```

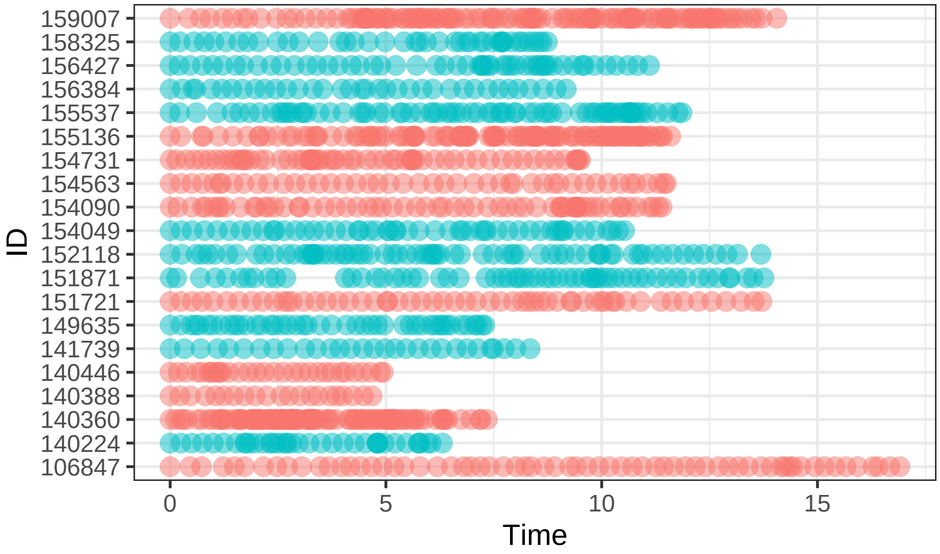


5. Make progress, as able with the latex file of the paper.

Start drafting a description of the dataset using Moss and Rice's papers: one paragraph to describe the study and a second to include descriptive statistics.

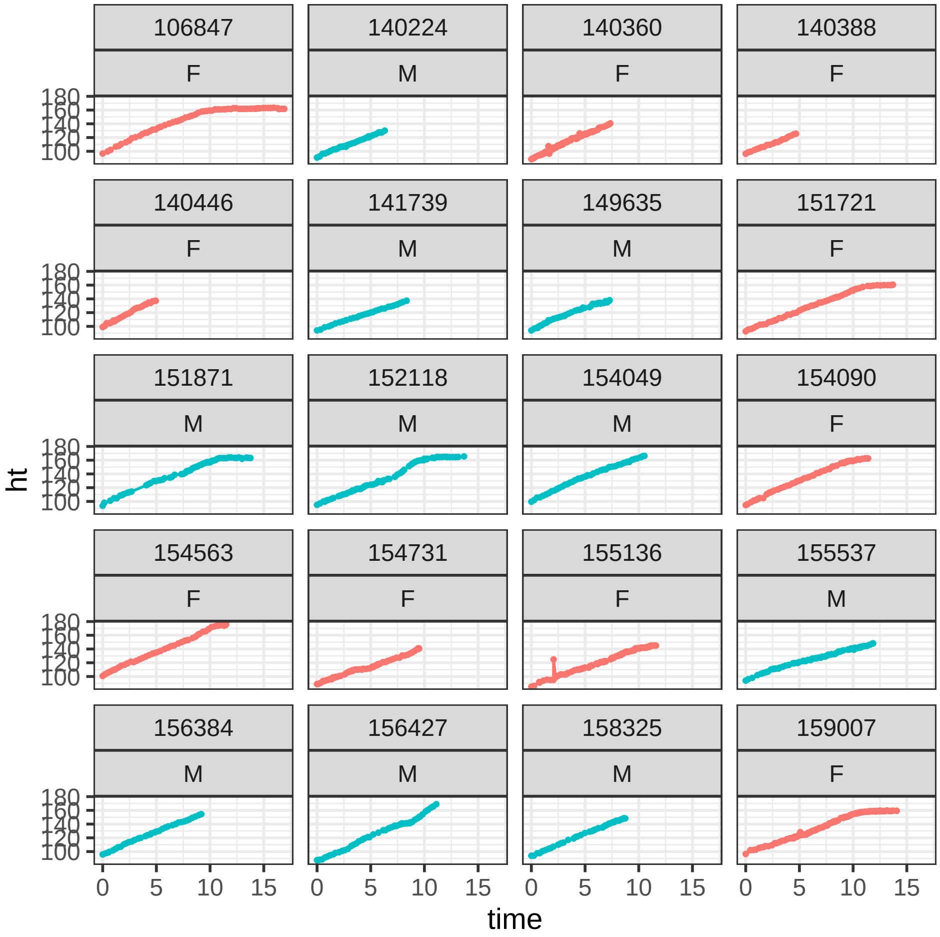
```
set.seed(555)
f_train <- train %>%
  filter(sex == "F")

m_train <- train %>%
  filter(sex == "M")

sub_id <- c(sample(f_train$id, size = 10, replace = FALSE),
            sample(m_train$id, size = 10, replace = FALSE))

subtrain <- train %>%
  filter(id %in% sub_id)
```

```
abacus0 <- ggplot(subtrain,
       aes(y = as.factor(id), x = time,
           group = sex,
           fill = sex, color = sex)) +
  geom_point(alpha = 0.5, size = 3) +
  theme_bw() +
  theme(legend.position = "none") +
  xlab("Time") +
  ylab("ID")

abacus0
```



```
ggsave("figure/abacus.png")
```

```
ind_plot <-
  ggplot(subtrain,
         aes(y = ht,
             x = time,
             group = id,
             col = sex)) +
  geom_point(size = 0.5) +
  geom_line() +
  theme_bw() +
  theme(legend.position = "none") +
  facet_wrap(id ~ sex, nrow = 5)

ind_plot
```



```r
ggsave("figure/sample_plot.png")
```

Dynamic Predictions in Bayesian Functional Joint Models for Longitudinal and Time-to-Event Data: An Application to Alzheimer’s Disease