skewBART for real data analysis (GAAD)

Seungha Um

September 23, 2022

Introduction

This vignette demonstrates how to fit the skewBART and multivariate skewBART models using data from the Type-2 diabetic Gullah-speaking African Americans (GAAD) study. This dataset is included in the package, and more details on this application can be found in Um, Linero, Sinha, and Bandyopadhyay (2022+, Statistics in Medicine). We will use a small number of MCMC iterations and trees to reduce computation times; these should instead be set to (say) 200 trees and 5000 iterations.

We begin by loading the required packages:

```
## Load library
library(tidyverse) # Load the tidyverse, mainly for ggplot
#> -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
#> v ggplot2 3.3.6      v purrr   0.3.4
#> v tibble  3.1.8      v dplyr   1.0.10
#> v tidyr   1.2.0      v stringr 1.4.1
#> v readr   2.1.2      v forcats 0.5.2
#> -- Conflicts ------------------------------------------ tidyverse_conflicts() --
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
library(kableExtra) # For fancy tables
#>
#> Attaching package: 'kableExtra'
#>
#> The following object is masked from 'package:dplyr':
#>
#>     group_rows
library(skewBART) # Our package
#> Loading required package: Rcpp
library(zeallot) # For the %<-% operator, used when generating data
options(knitr.table.format = 'markdown')
```

Univariate skewed response

There are two responses in the GAAD dataset; the mean clinical attachment level (CAL) and the mean periodontal pocket depth (PPD). We begin with a univariate analysis of the CAL response. The data can be accessed in the package by running the commands

```
data(GAAD)
Y <- GAAD$subCAL
X <- GAAD %>% dplyr::select(Age, Female, Bmi, Smoker, Hba1cfull)
```

As in the `Vignette` vignette (see `browseVignettes("skewBART")`), we first create hyperparameters and set the MCMC options:

```
hypers <- UHypers(X, Y, num_tree = 20)
opts <- UOpts(num_burn = 250, num_save = 250)
```

We will use the `skewBART_pd` function to make partial dependence plots in order to examine the marginal effects of some groups of covariates. The following specifies the values (in the form of a `data.frame`) to compute the marginal effect at:

```
x_grid <- expand.grid(Hba1cfull = seq(5, 16.4, length = 20),
                      Smoker = c(0,1), Female = c(0,1))
```

We then fit the model using the `skewBART_pd` command:

```
set.seed(77777)
fitted_skewbart <- skewBART_pd(X, Y, vars = c("Hba1cfull", "Smoker", "Female"),
                               x_grid = x_grid,
                               hypers = hypers, opts = opts)
#> Finishing iteration 100 of 500   Finishing iteration 200 of 500  Finishing iteration 300 of 500  Fin
```

Later, we will also need the fit of the model to PD:

```
set.seed(77777)
PD <- GAAD$subPD
hypers <- UHypers(X, PD, num_tree = 20)
fitted_skewbart_pd <- skewBART_pd(X, PD, vars = colnames(x_grid),
                                  x_grid = x_grid,
                                  hypers = hypers, opts = opts)
#> Finishing iteration 100 of 500   Finishing iteration 200 of 500  Finishing iteration 300 of 500  Fin
```

We can compare this fit with a model fit on the log scale (for simplicity, we don't compute the partial dependence here):

```
set.seed(77777)
hypers <- UHypers(X, log(Y), num_tree = 20)
fitted_skewbart_log <- skewBART(X, log(Y), X, hypers, opts)
#> Finishing iteration 100 of 500   Finishing iteration 200 of 500  Finishing iteration 300 of 500  Fin
```

We might then compare the two fits by LPML; for the original model, this is

```
fitted_skewbart$loo$estimates["elpd_loo",1]
#> NULL
```

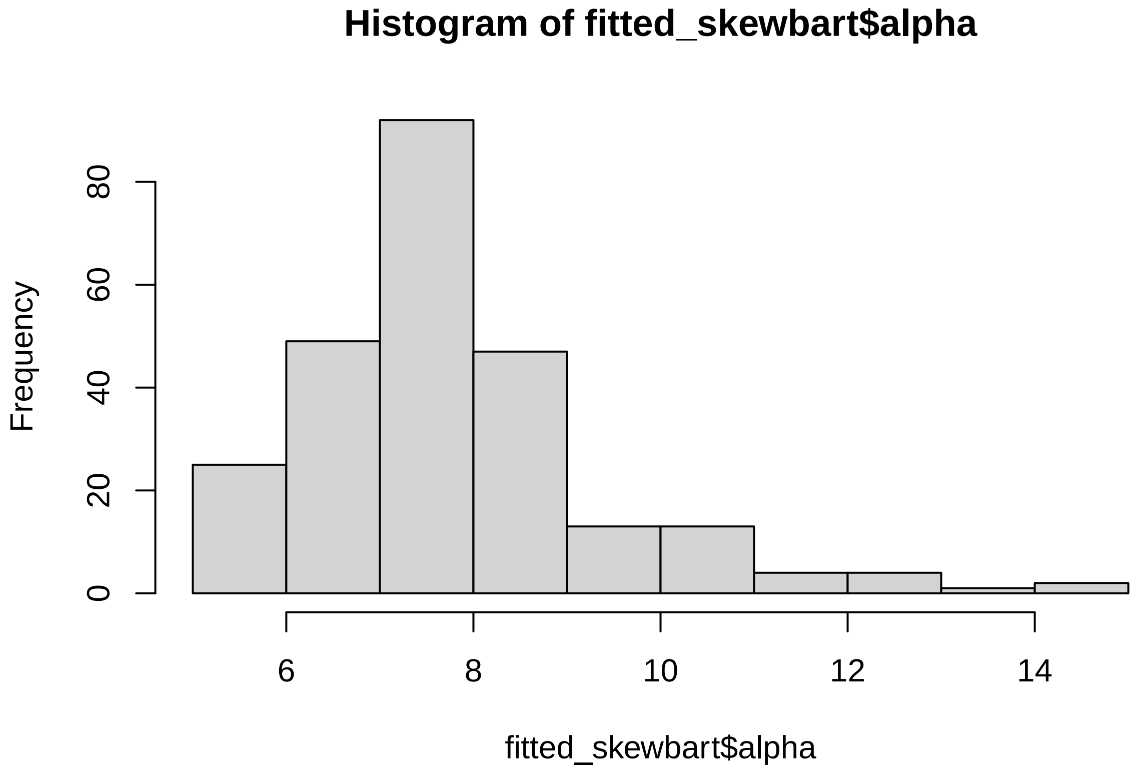
while for the log model we need to take into account the Jacobian of the log transformation $\log(Y)$ to give a fair comparison:

```
fitted_skewbart_log$loo$estimates["elpd_loo",1] - sum(log(Y))
#> numeric(0)
```

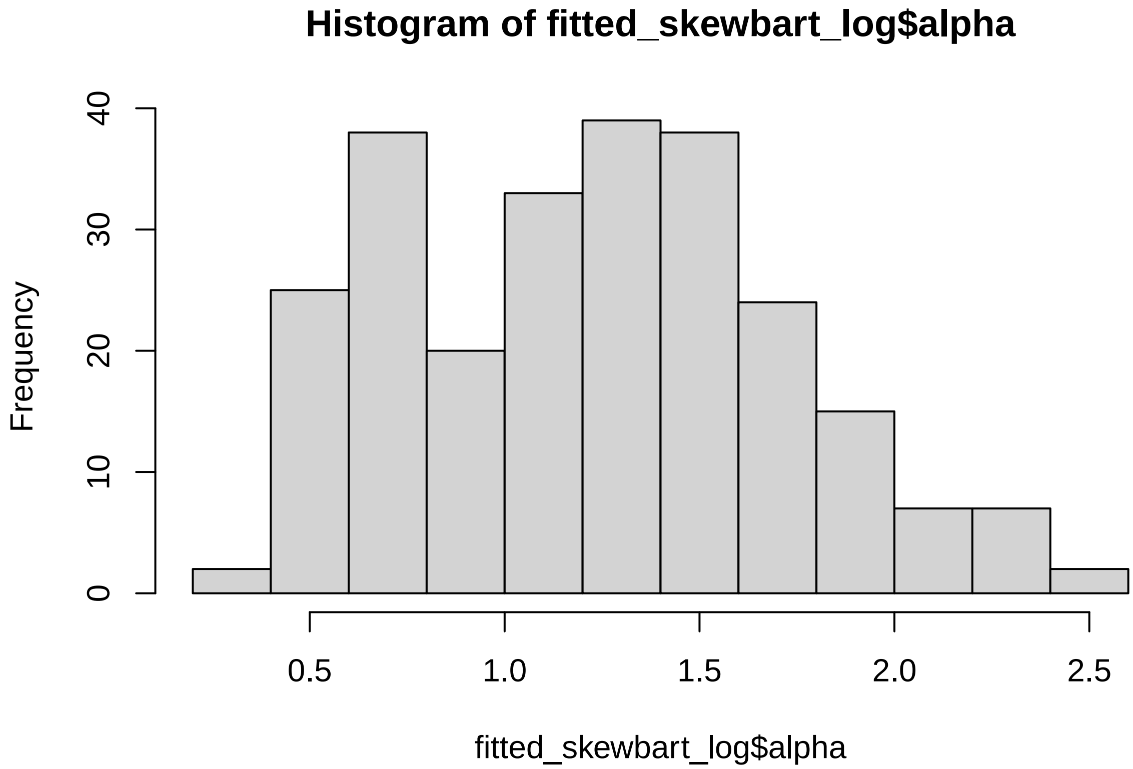
We see that the model with the log transformation performs better overall.

We next examine the posterior of the skewness α:

```
hist(fitted_skewbart$alpha)
```



```
hist(fitted_skewbart_log$alpha)
```



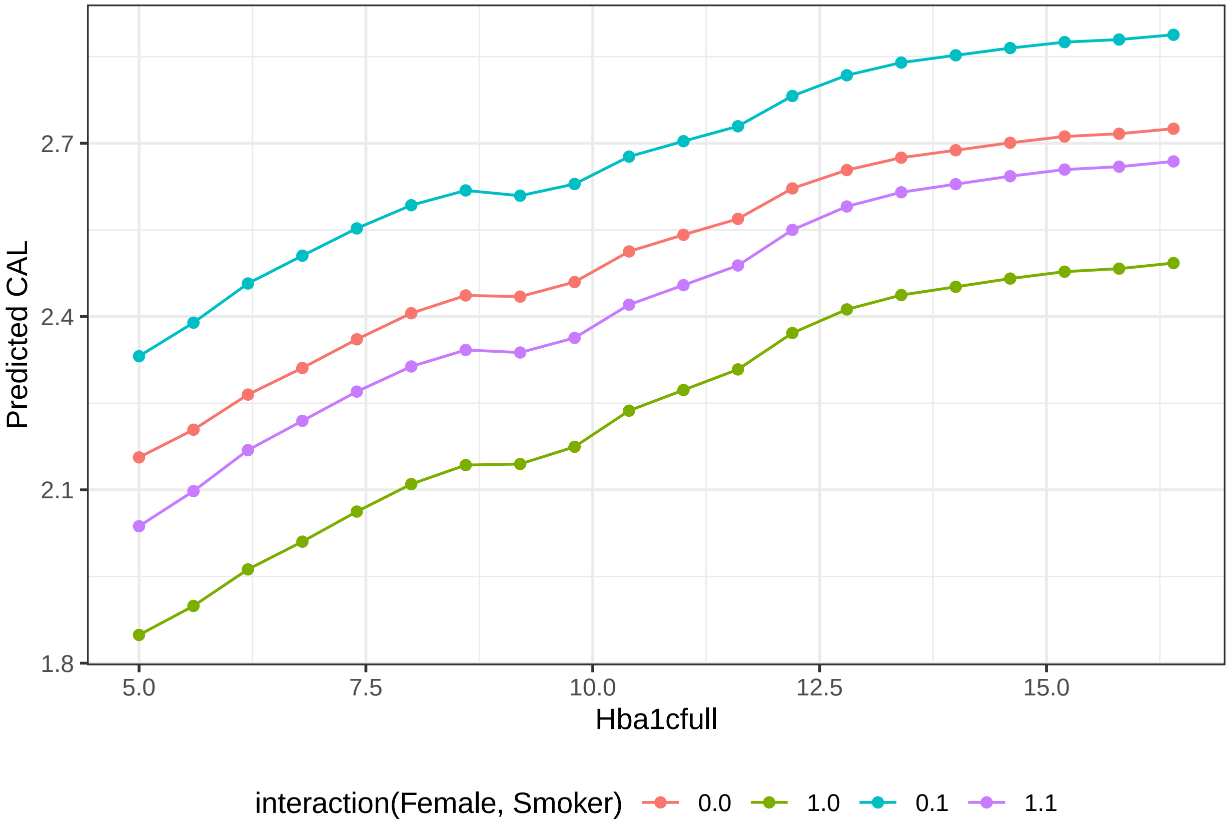
We see that, even after using the log transformation, the data still suggests that we should use a skew-normal model over a normal model for the error (α is concentrated away from 0).

Next, we examine the partial dependence plots to see (i) how Hba1c affects CAL and (ii) how this effect is moderated by gender and smoking. The fitted model contains both the samples and summary statistics (i.e., the posterior mean and the 2.5th and 97.5th percentiles) of the partial dependence function:

```
head(fitted_skewbart$partial_dependence_samples)
#>   Iteration Hba1cfull Smoker Female     f_hat    y_hat
#> 1         1         5      0      0 0.9419774 2.167561
#> 2         2         5      0      0 0.9473918 2.163663
#> 3         3         5      0      0 1.0506339 2.280249
#> 4         4         5      0      0 1.0022178 2.200073
#> 5         5         5      0      0 0.9858477 2.194963
#> 6         6         5      0      0 1.1545160 2.327019
head(fitted_skewbart$partial_dependence_summary)
#> # A tibble: 6 x 9
#>   Hba1cfull Smoker Female f_hat_mean f_hat_025 f_hat_975 y_hat~1 y_hat~2 y_hat~3
#>       <dbl>  <dbl>  <dbl>      <dbl>     <dbl>     <dbl>   <dbl>   <dbl>   <dbl>
#> 1       5        0      0      0.847     0.522     1.11     2.16    1.87    2.55
#> 2       5        0      1      0.540     0.232     0.780    1.85    1.55    2.14
#> 3       5        1      0      1.02      0.714     1.32     2.33    2.03    2.70
#> 4       5        1      1      0.728     0.381     1.01     2.04    1.65    2.29
#> 5       5.6      0      0      0.895     0.636     1.10     2.20    1.97    2.56
#> 6       5.6      0      1      0.590     0.342     0.764    1.90    1.71    2.13
#> # ... with abbreviated variable names 1: y_hat_mean, 2: y_hat_025, 3: y_hat_975
```

Here, $\widehat{f}$ is the BART-modeled function while $\widehat{Y}$ is the predicted value after accounting for the fact that the errors are not mean 0. We use this to plot the partial dependence:

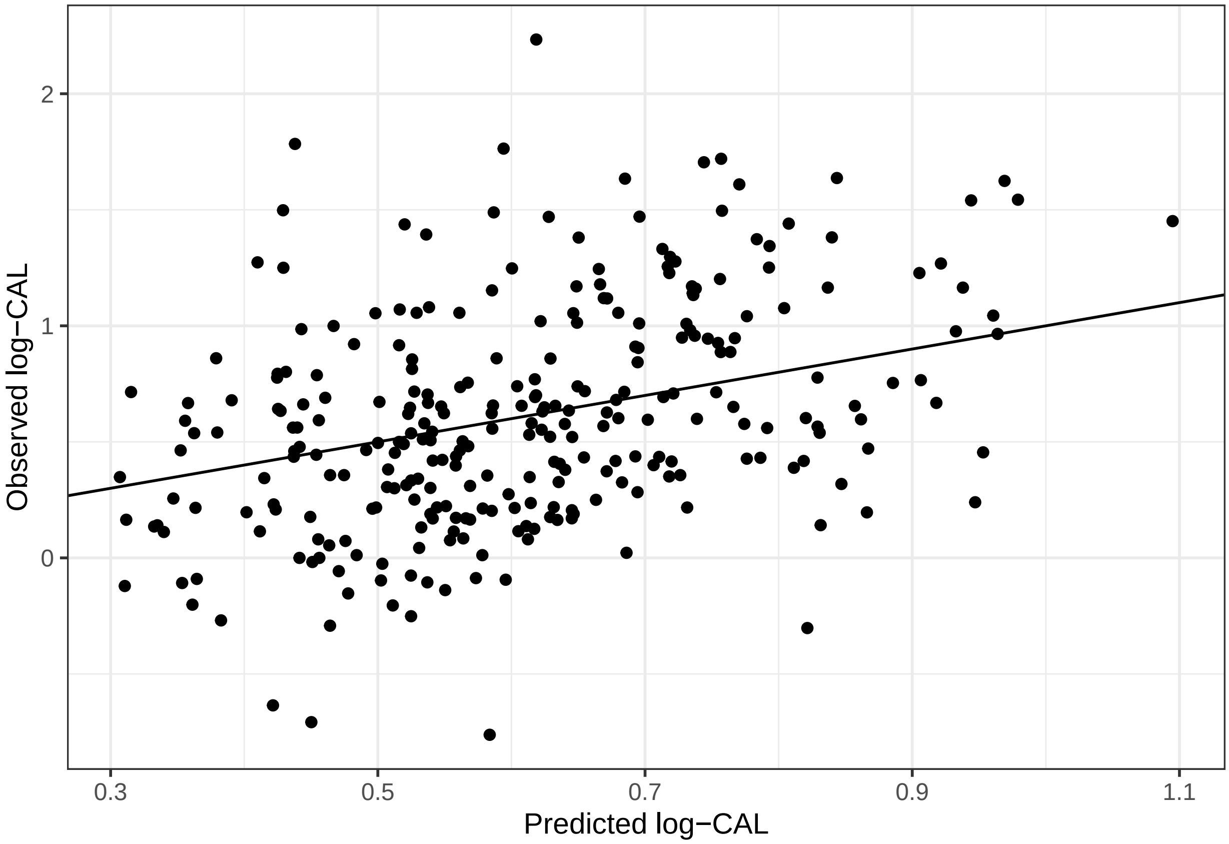
```
ggplot(fitted_skewbart$partial_dependence_summary,
      aes(x = Hba1cfull, y = y_hat_mean, color = interaction(Female, Smoker))) +
  geom_point() + geom_line() + theme_bw() + ylab("Predicted CAL") +
  theme(legend.position = "bottom")
```



The estimated effects are not quite homogeneous, with female smokers starting closer to female non-smokers for low values of Hba1c and ending up nearly equal to male non-smokers for high values of Hba1c. Overall, males have higher values of CAL.

Next, we display log-CAL against its predicted values:

```
qplot(fitted_skewbart_log$y_hat_train_mean, log(Y)) +
  geom_abline(slope = 1, intercept = 0) + theme_bw() +
  xlab("Predicted log-CAL") + ylab("Observed log-CAL")
```



We see that the model does a good job capturing the relationship between log-CAL and the predictors.

Multivariate skewed response

Next, we fit the multivariate skewBART model to both the CAL and PPD. The outcome `Y` is now a 2 by 2 matrix:

```
X <- GAAD %>% select("Age", "Female", "Bmi", "Smoker", "Hba1cfull") %>% as.matrix()
Y <- GAAD %>% select(subCAL, subPD) %>% as.matrix()
```

We then build the hyperparameter/MCMC objects:

```
hypers <- Hypers(X = X, Y = Y, num_tree = 20)
opts <- Opts(num_burn = 50, num_save = 50, num_print = 10)
```

We can fit the model, creating partial dependence plots as before, using the `MultiskewBART_pd` function:

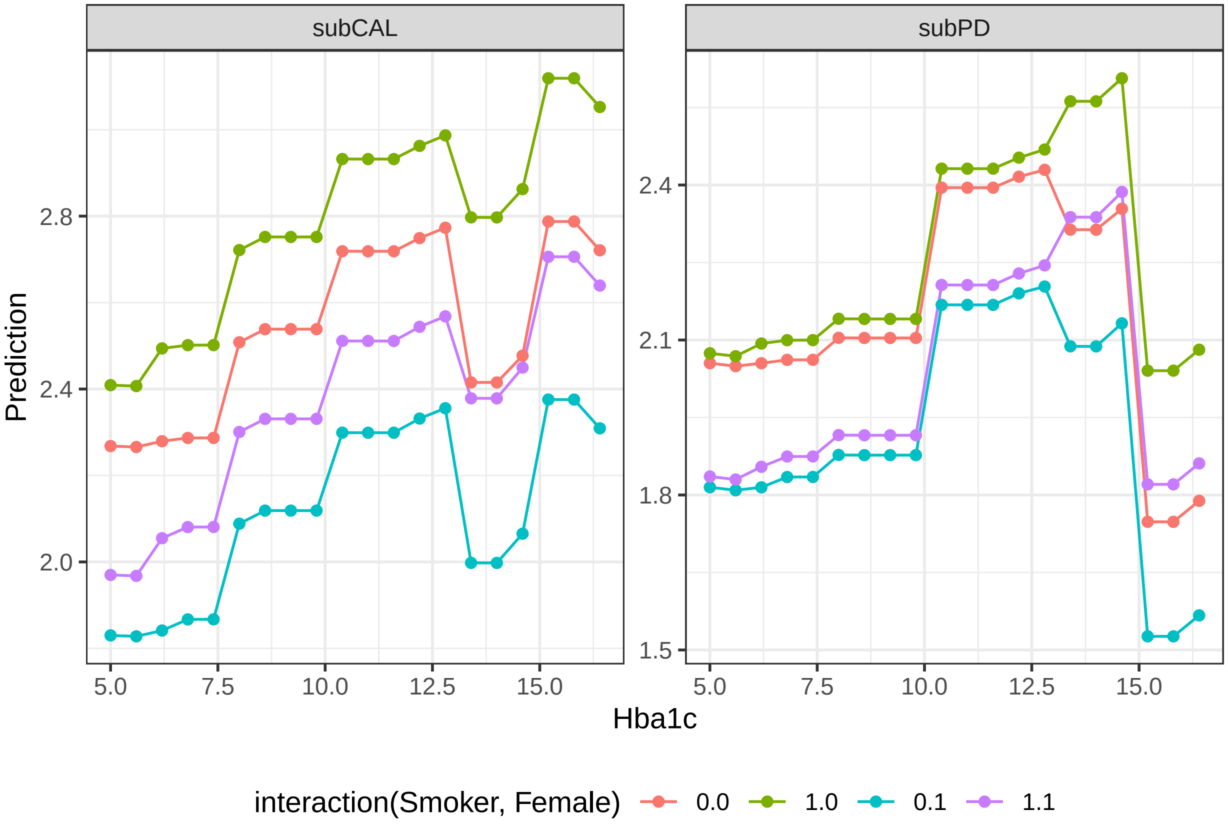
```
set.seed(77777)
fitted_Multiskewbart <- MultiskewBART_pd(
  X = X, Y = Y, vars = colnames(x_grid), x_grid = x_grid, hypers = hypers,
  opts = opts
)
#> Finishing iteration 10 of 100Finishing iteration 20 of 100Finishing iteration 30 of 100Finishing ite
```

For the purposes of model comparison we can again look at the LPML:

```
fitted_Multiskewbart$loo
#> NULL
```

And lastly we can construct our partial dependence plots:

```
head(fitted_Multiskewbart$partial_dependence_summary)
#> # A tibble: 6 x 10
#>   Hba1cf~1 Smoker Female f_hat~2 f_hat~3 f_hat~4 y_hat~5 y_hat~6 y_hat~7 outcome
#>      <dbl>  <dbl>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>
#> 1      5         0      0    1.48   1.17     1.75    2.27    1.96    2.53 subCAL
#> 2      5         0      1    1.04   0.807    1.21    1.83    1.60    1.99 subCAL
#> 3      5         1      0    1.62   1.11     1.91    2.41    1.90    2.70 subCAL
#> 4      5         1      1    1.18   0.724    1.42    1.97    1.51    2.21 subCAL
#> 5      5.6       0      0    1.48   1.17     1.75    2.27    1.96    2.54 subCAL
#> 6      5.6       0      1    1.04   0.807    1.21    1.83    1.59    1.99 subCAL
#> # ... with abbreviated variable names 1: Hba1cfull, 2: f_hat_mean,
#> #   3: f_hat_025, 4: f_hat_975, 5: y_hat_mean, 6: y_hat_025, 7: y_hat_975
fitted_Multiskewbart$partial_dependence_summary %>%
  ggplot(aes(x = Hba1cfull, y = y_hat_mean,
             color = interaction(Smoker, Female))) +
  geom_line() + geom_point() + facet_wrap(.~outcome, scales = "free_y") +
  theme_bw() +theme(legend.position = "bottom") + ylab("Prediction") +
  xlab("Hba1c")
```

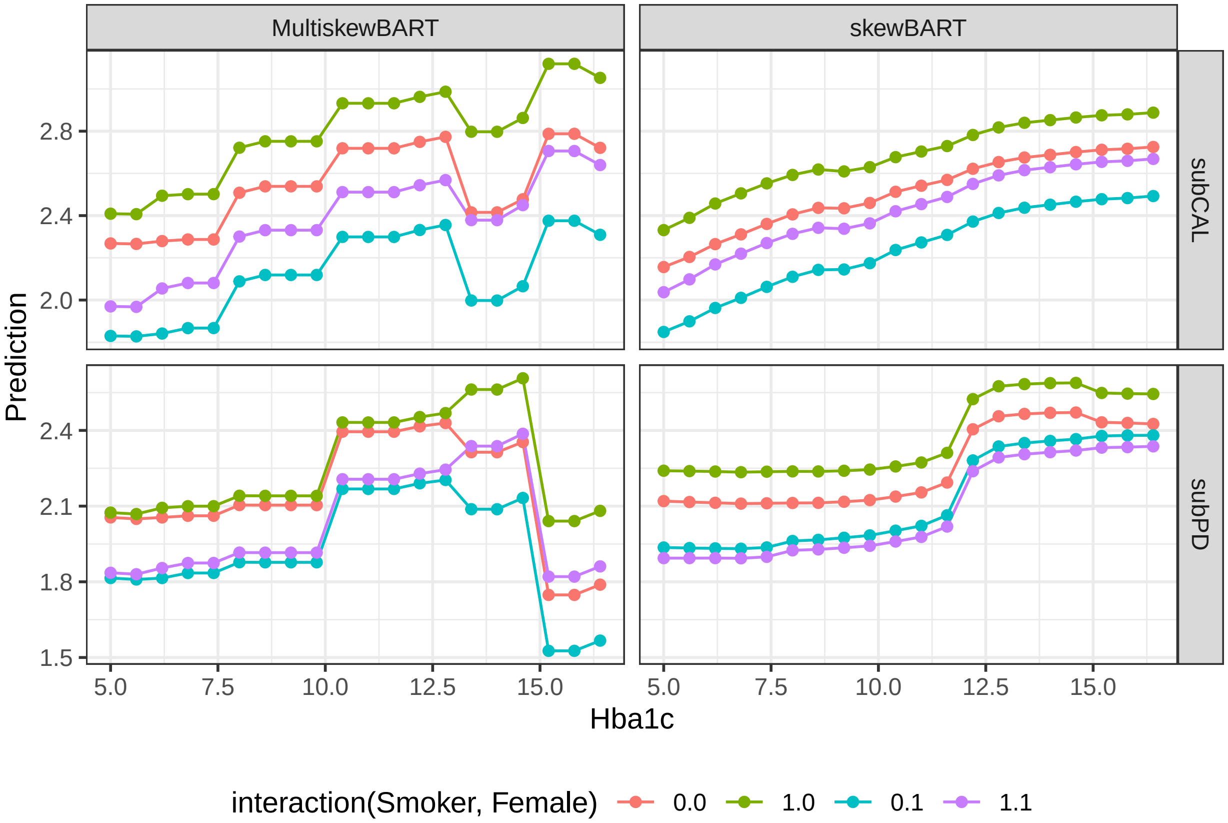


For comparison, let's also look at the same plots using univariate fits to the data.

```
skew_pd_cal <- fitted_skewbart$partial_dependence_summary %>%
  mutate(outcome = "subCAL", method = "skewBART")
skew_pd_pd <- fitted_skewbart_pd$partial_dependence_summary %>%
  mutate(outcome = "subPD", method = "skewBART")
mskew_pd <- fitted_Multiskewbart$partial_dependence_summary %>%
  mutate(method = "MultiskewBART")
```

```r
skew_pd <- rbind(skew_pd_cal, skew_pd_pd, mskew_pd)

skew_pd %>%
  ggplot(aes(x = Hba1cfull, y = y_hat_mean,
             color = interaction(Smoker, Female))) +
  geom_line() + geom_point() + facet_grid(outcome~method, scales = "free_y") +
  theme_bw() +theme(legend.position = "bottom") + ylab("Prediction") +
  xlab("Hba1c")
```



The results for MultiskewBART and skewBART are similar, with skewBART being much smoother. When a larger number of iterations are used to fit the MultiskewBART model, the results are otherwise quite similar.