

Biostat 276 Project 1

Kelly Li

1/29/2022

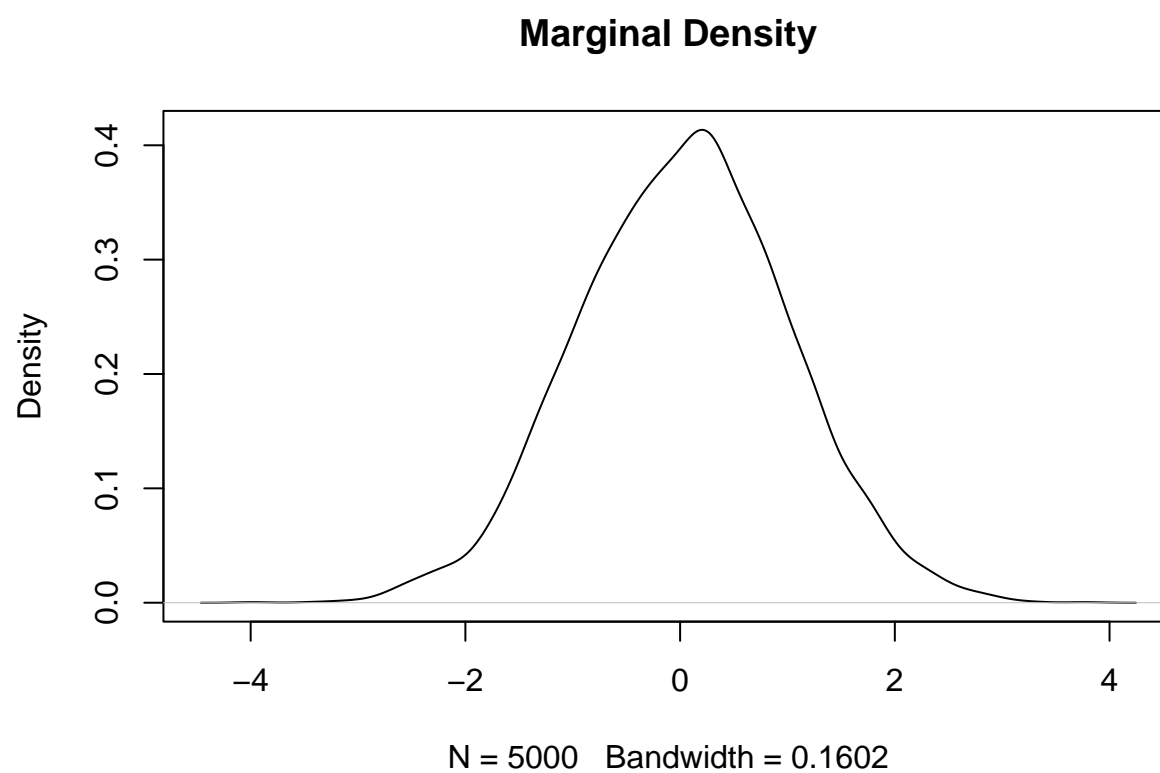
Sampling from the Banana Distribution

a)

Bayesian Adaptive Lasso

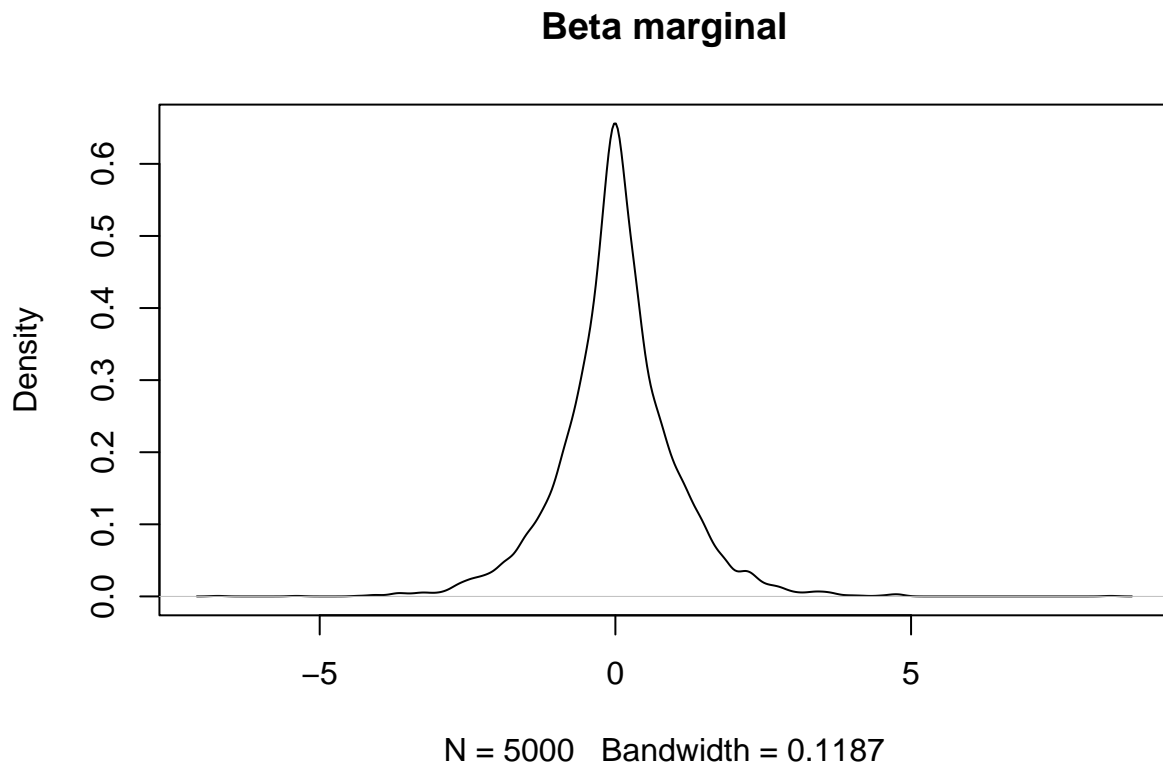
#a)

```
sima <- rnorm(5000, 0, 1)
plot(density(sima), main = "Marginal Density")
```



b)

```
lambda2 <- 2
tau2 <- rgamma(5000, shape = 1, rate = lambda2/2)
simb <- rnorm(5000, 0, sqrt(tau2))
plot(density(simb), main = "Beta marginal")
```

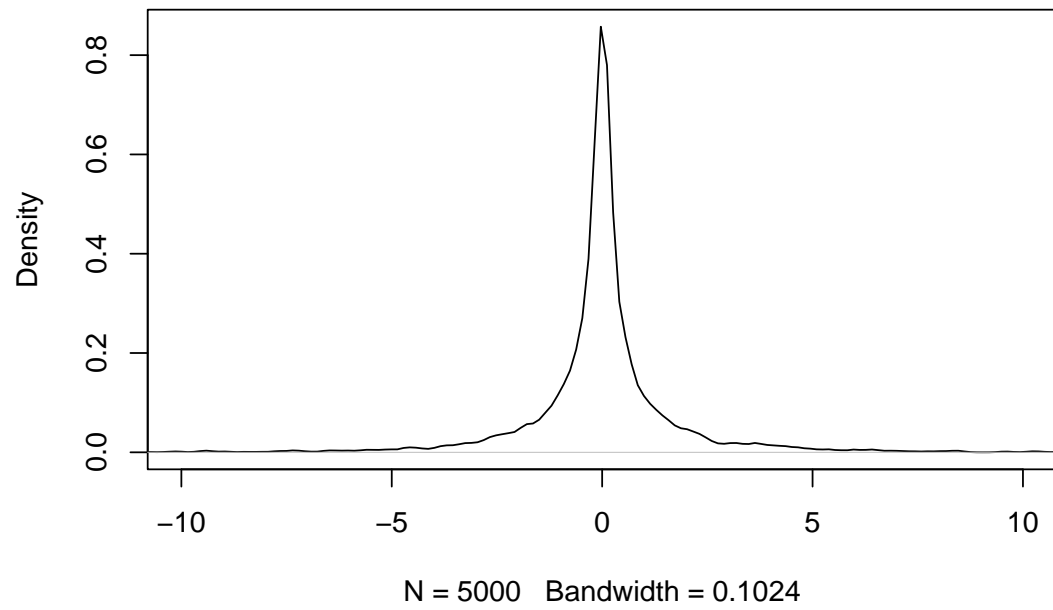


c)

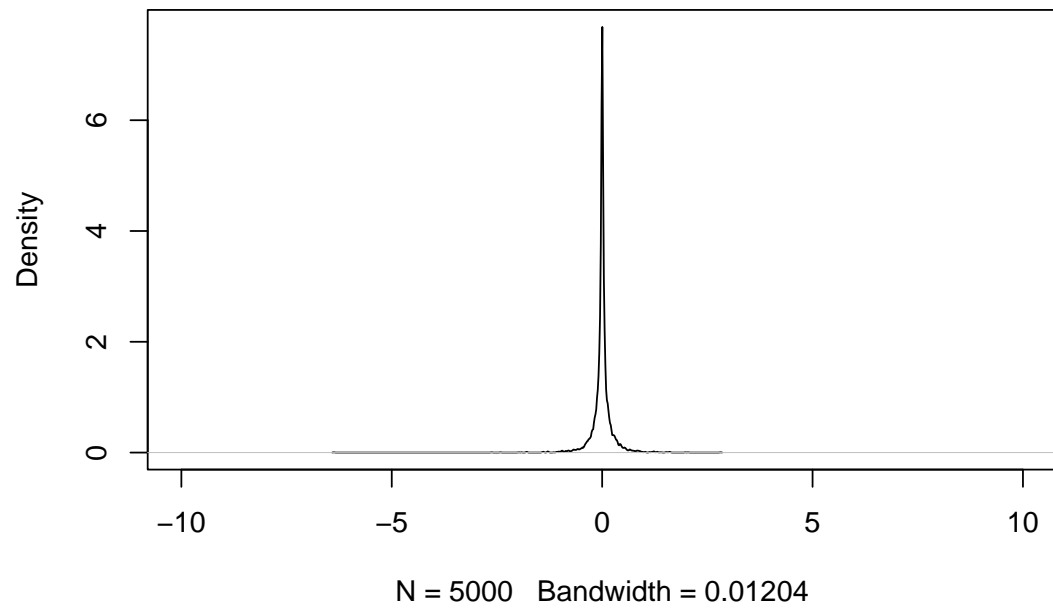
```
bvec <- c(1, 8, 20, 10000)
marginalplot <- function(n, b){
  lambda <- 1/rgamma(n, 1, b)
  tau2 <- rgamma(n, shape = 1, rate = lambda^2/2)
  sim <- rnorm(n, 0, sqrt(tau2))
  plot <- plot(density(sim),
               main = paste0("Beta marginal, b = ", b),
               xlim = c(-10, 10))
  save_plot <- recordPlot()
  return(save_plot)
}

plots <- lapply(bvec, marginalplot, n = 5000)
```

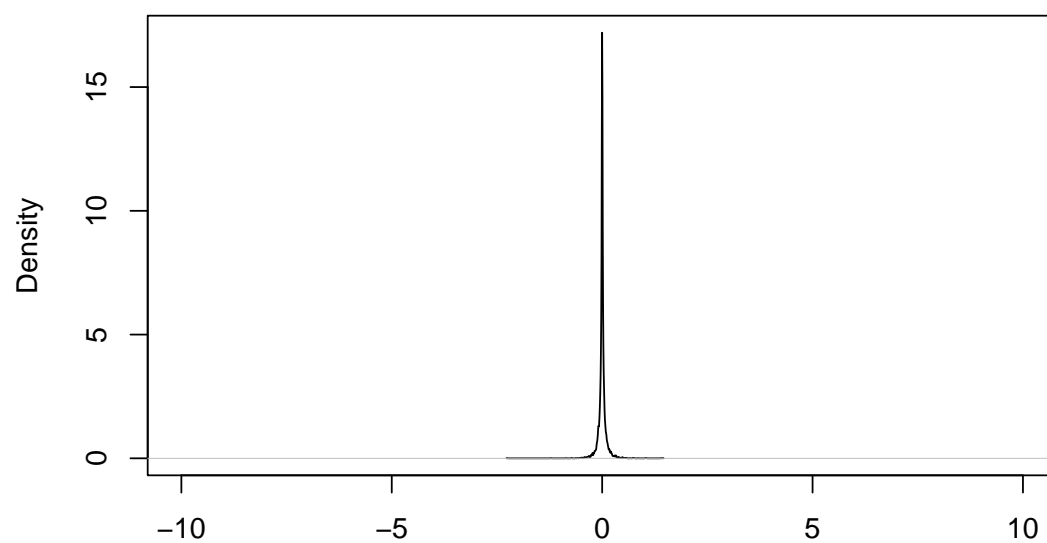
Beta marginal, $b = 1$



Beta marginal, $b = 8$

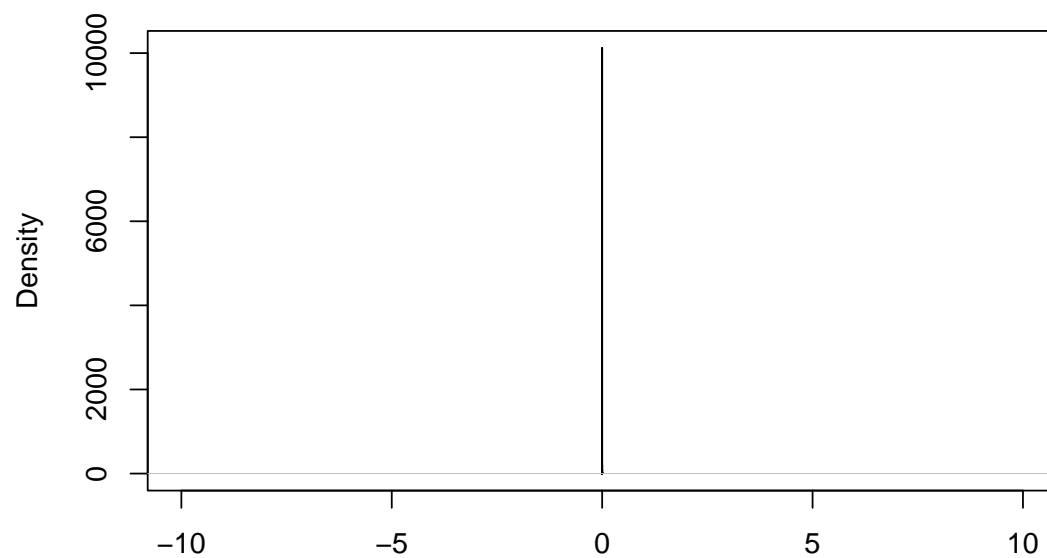


Beta marginal, b = 20



N = 5000 Bandwidth = 0.005061

Beta marginal, b = 10000



N = 5000 Bandwidth = 9.647e-06

d)

e)

I will implement a Gibbs Sampler algorithm to sample from the posterior distributions.

```
gibbs <- function(y = y, X = Xdat,
                 lambda2, tau2,
                 n.sim = 1000, burn = 0.1,
                 a = 1, b = 1, beta, fixlambda = FALSE){
  #Chain information
  n.total <- n.sim*(1.0 + burn)
  n.burn <- n.sim*burn

  #Initializing matrices
  betamu.out <- matrix(NA, n.sim, 10)
  beta.out <- matrix(NA, n.sim, 10)
  sigma2.out <- c()

  #Data and Parameters
  n <- length(y)
  p <- ncol(X)
  XtX <- t(X) %*% X

  for(i in 1:n.total){
    rss <- t(y - X %*% beta)%*(y - X %*% beta)
    #lambda2
    if(fixlambda == TRUE){
      lambda2 <- lambda2
    } else {
      lambda2 <- rgamma(1, shape = a, rate = sum(tau2)/2 + b)
    }
    #tau2
    for(a in 1:length(beta)){
      tau2[a] <- rinvgauss(1, sqrt(lambda2)/sqrt(beta[a]^2), lambda2)
    }
    tau2 <- 1/tau2
    #sigma2
    shape <- 0.1
    scale <- rss/2 + 0.1
    sigma2 <- 1/rgamma(1, shape = shape, scale = scale)
    #beta
    betavar <- solve(XtX/sigma2 + solve(diag(tau2)))
    betamu <- betavar %*% t(X) %*% y/sigma2
    beta <- rmvnorm(n=1, mean = betamu, sigma = betavar) %>% t()
    if(i > n.burn){
      i1 <- i - n.burn
      betamu.out[i1, ] <- betamu
      beta.out[i1, ] <- beta
      sigma2.out[i1] <- sigma2
    }
  }
  return(list(beta = beta.out, sigma2 = sigma2.out))
}

data("diabetes")
```

My MCMC	Glmnet
-10.01230	-9.057331
-239.81901	-238.879817
519.83970	520.885704
324.39005	323.449275
-792.18023	-680.104304
476.74301	390.311899
101.04347	48.691747
177.06407	159.307982
751.27797	710.278264
67.62555	67.555464

```
Xdat <- diabetes$x
y <- diabetes$y
betahat <- solve(t(Xdat) %*% Xdat) %*% t(Xdat) %*% y
out <- gibbs(y = y, X = Xdat, n = 1000, burn = 0.1,
            a = 1, b = 1, beta = betahat, lambda2 = 1, tau2 = rep(1, 10))

coeffun <- apply(out$beta, 2, function(x){quantile(x,c(0.16,0.5,0.84))})
coefme <- coeffun[2,]

fit.glm <- glmnet(Xdat, y, intercept = FALSE)
coefglm <- coef(fit.glm, s = min(fit.glm$lambda))
compare <- tibble("My MCMC" = coefme, "Glmnet" = matrix(coefglm[-1]))

kable(compare) %>% kable_styling(full_width = F)
```

f

```
lambdas <- exp(seq(-4, 4, 0.1))^2
out <- lapply(lambdas, function(x)gibbs(y = y, X = Xdat,
                                       n.sim = 1000, burn = 0.1,
                                       a = 1, b = 1, beta = rep(0, 10),
                                       lambda2 = x, tau2 = rep(1, 10),
                                       fixlambda = TRUE))
betas <- lapply(out, `[`, c("beta"))
betas <- lapply(betas, sapply, colMedians)

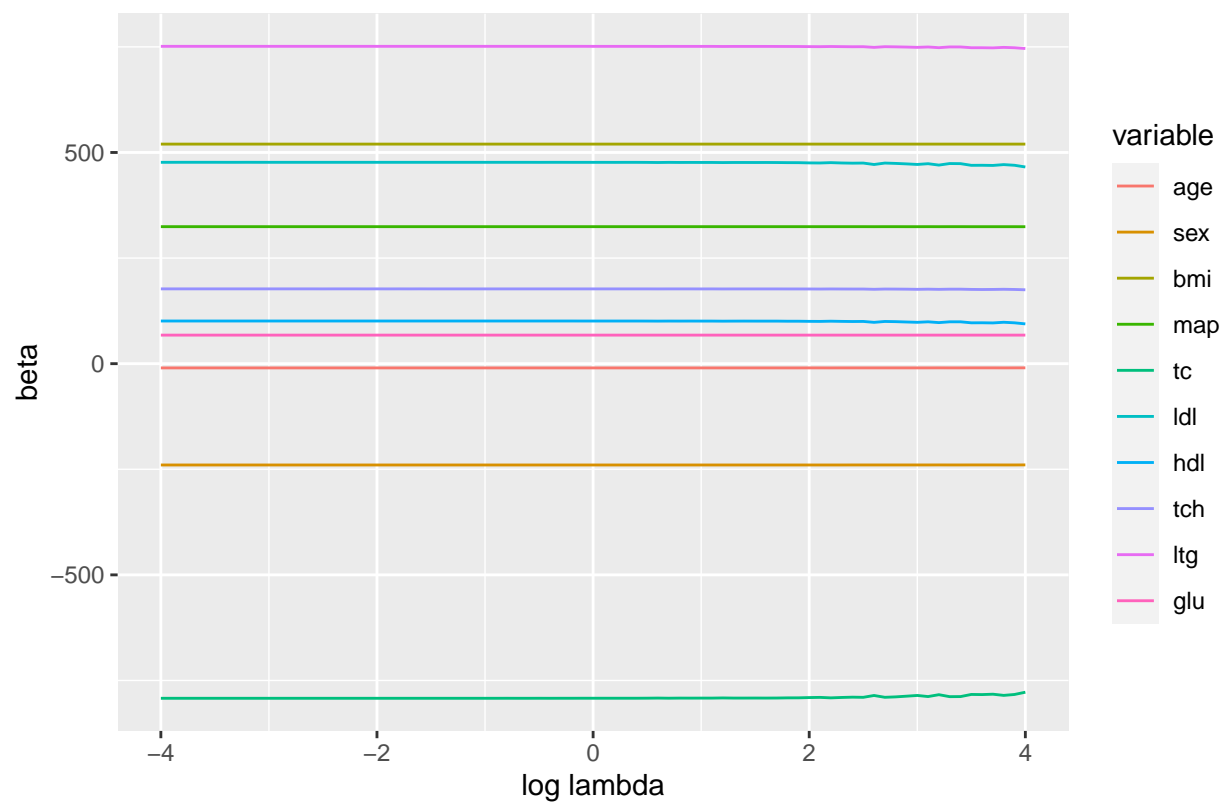
df <- do.call("cbind", betas)
rownames(df) <- colnames(Xdat)
df <- t(df) %>% as.data.frame() %>% melt()

## No id variables; using all as measure variables

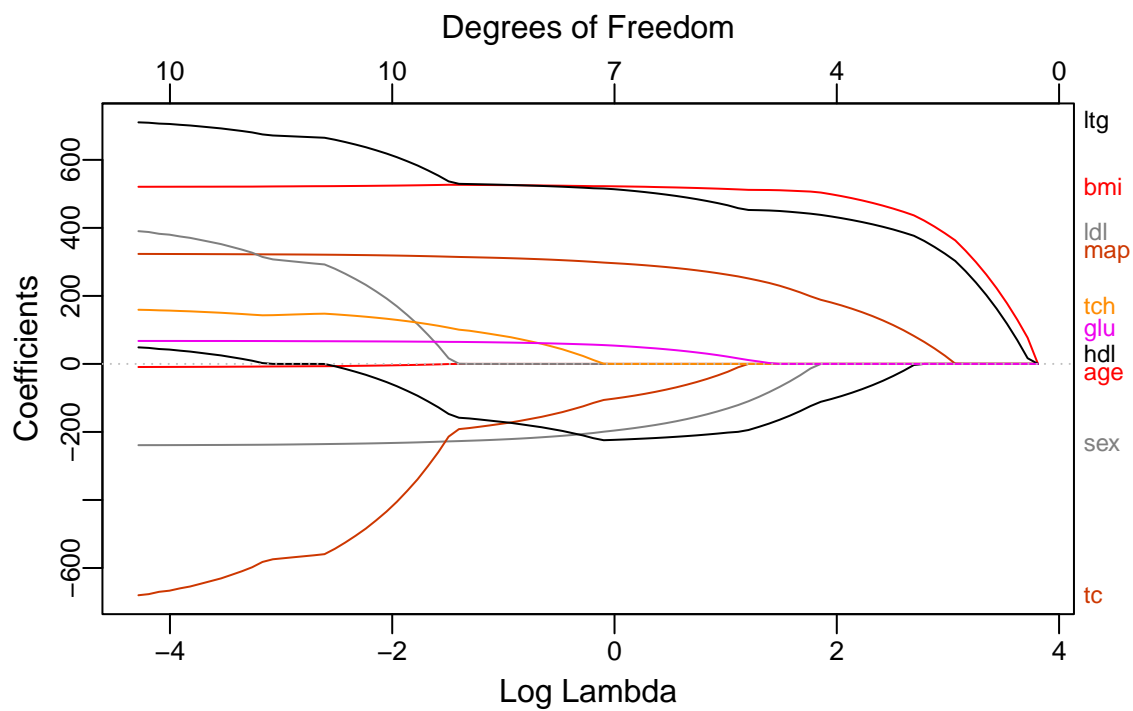
df$lambda <- rep(seq(-4, 4, 0.1), 10)

ggplot(data = df, aes(lambda, value, color = variable)) +
  geom_line() +
  ggtitle("Custom MCMC path") + xlab("log lambda") + ylab("beta")
```

Custom MCMC path



```
plot_glmnet(fit.glm, xvar = "lambda")
```



g

```
df.ab <- data.frame(a = c(1, 5, 10, 1, 5, 10, 1, 5, 10),
                    b = c(1, 1, 1, 5, 5, 5, 10, 10, 10))
out <- apply(df.ab, 1, function(df.ab) gibbs(y = y, X = Xdat,
                                             n.sim = 1000, burn = 0.1,
                                             a = df.ab[1], b = df.ab[2], beta = betahat,
                                             lambda2 = 1, tau2 = rep(1, 10)))
betas <- lapply(out, `[`, c("beta"))
betas <- lapply(betas, sapply, colMedians)
df <- do.call("cbind", betas)
colnames(df) <- paste0("(", df.ab$a, ", ", df.ab$b, ")")
rownames(df) <- colnames(Xdat)
df %>% kable() %>% kable_styling(full_width = F)
```


	(1, 1)	(5, 1)	(10, 1)	(1, 5)	(5, 5)	(10, 5)	(1, 10)	(5, 10)	(10, 10)
age	-10.01224	-10.01214	-10.01239	-10.01215	-10.01216	-10.01215	-10.01232	-10.01199	-10.01215
sex	-239.81882	-239.81888	-239.81875	-239.81866	-239.81878	-239.81895	-239.81886	-239.81899	-239.81888
bmi	519.83964	519.83940	519.83930	519.83964	519.83929	519.83958	519.83964	519.83949	519.83930
map	324.39011	324.39035	324.39044	324.39024	324.39025	324.39026	324.39031	324.39011	324.39024
tc	-792.17625	-792.17902	-792.17963	-792.17917	-792.17929	-792.17916	-792.18010	-792.18170	-792.17917
ldl	476.73979	476.74210	476.74347	476.74217	476.74146	476.74254	476.74272	476.74407	476.74217
hdl	101.04214	101.04232	101.04289	101.04334	101.04333	101.04320	101.04293	101.04404	101.04334
tch	177.06270	177.06334	177.06262	177.06366	177.06354	177.06320	177.06360	177.06375	177.06334
ltg	751.27646	751.27665	751.27809	751.27722	751.27825	751.27775	751.27787	751.27829	751.27722
glu	67.62499	67.62548	67.62532	67.62528	67.62526	67.62513	67.62546	67.62535	67.62528