

Week 7 HWK

Ross Walker, Mike Kozel, Qi Hao

March 21, 2018

Plot 1

Replicate the last plot from <https://fivethirtyeight.com/features/what-the-world-thinks-of-trump/>

Note: Fonts were shrunk to appear clean on the knitted HTML rather than in RStudio output.

```
suppressMessages(library(reshape2))
suppressMessages(library(ggplot2))
suppressMessages(library(dplyr))

urls <- paste0("https://raw.githubusercontent.com/fivethirtyeight/data/master/trump-world-trust/TRUMPWORLD")

datalist = lapply(urls, function(x){read.csv(file = x, header = T, stringsAsFactors = F)})
names(datalist) <- c('climate_change', 'wall', 'Iran', 'trade', 'Muslim_travel')

regions <- list(
  `Europe and Russia` = c('France', "Germany", "Greece", "Hungary", "Italy", "Netherlands", "Poland", "Russia", "U.K.", "U.S."),
  `Middle East and Africa` = c("Israel", "Jordan", "Lebanon", "Tunisia", "Turkey", "Ghana", "Kenya", "Nigeria", "South Africa"),
  `South America and Mexico` = c("Argentina", "Brazil", "Chile", "Colombia", "Mexico", "Peru", "Venezuela"),
  `Asia` = c("India", "Indonesia", "Japan", "Philippines", "South Korea", "Vietnam")
)

for(i in 1:length(datalist)){
  colnames(datalist[[i]])[2] <- names(datalist)[i]
  datalist[[i]] <- datalist[[i]][1:2]
}

main <- Reduce(function(x,y) {merge(x, y, by = 'country')}, datalist)
main <- main[-which(main$country %in% c("Canada", "Australia")), ]
main$region <- sapply(main$country, function(y){
  names(regions)[unlist(lapply(regions, function(x){
    return(y %in% x)
  }))]
}) %>%
  unlist

suppressMessages(gg <- melt(main))
gg$region <- as.factor(gg$region)
levels(gg$region) <- c("Asia", "Europe\nand Russia", "Middle East\nand Africa", "South America\nand Mexico", "U.S.\nand Canada")
levels(gg$variable) <- c("Withdraw from global climate change agreements", "Build a wall between the U.S. and Mexico", "Ban Muslims from entering the country", "Trade with Iran", "Trade with Russia")

ggplot(gg, aes(x = value, y = region, fill = factor(region))) +
  geom_point(shape = 21, size = 3, alpha = .5, aes(stroke = ifelse(country %in% c("Russia", "Germany", "Israel", "Turkey", "Brazil", "Mexico", "U.S.", "Canada"), "black", "grey"))) +
  geom_text(aes(label = ifelse(country %in% c("Russia", "Germany", "Israel", "Turkey", "Brazil", "Mexico", "U.S.", "Canada"), country, ""))) +
  geom_vline(xintercept = 0, color = "grey") +
  geom_vline(data = gg %>% group_by(variable) %>% summarise(avg = mean(value, na.rm = T)), aes(xintercept = avg)) +
  theme_light()
```

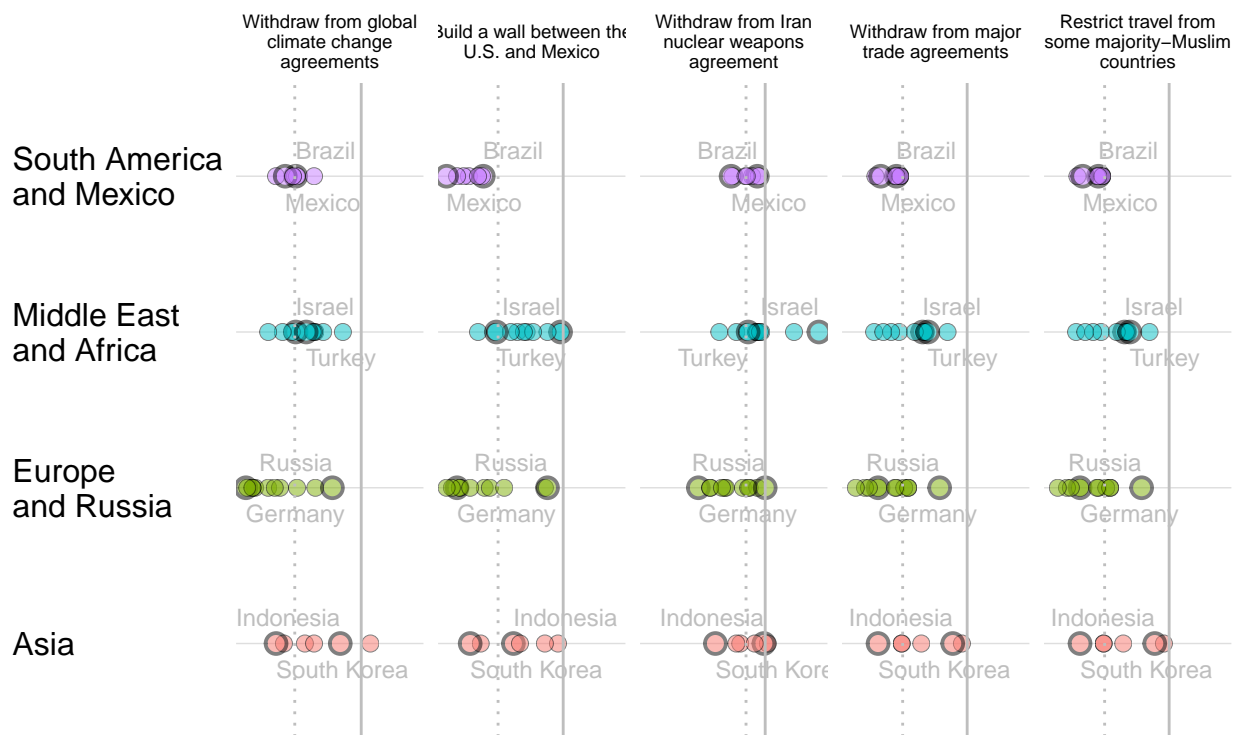
```

ggtitle(label = "What do people in other countries think of Trump's proposed policies?", subtitle = "Net approval rating for Trump's proposed policy to...")
facet_wrap(~variable, nrow = 1, labeller=label_wrap_gen()) +
theme(
  panel.grid.minor = element_blank(),
  panel.border = element_blank(),
  axis.ticks = element_blank(),
  axis.title = element_blank(),
  axis.text.x = element_blank(),
  axis.text.y = element_text(color = "black", size = 12, hjust = 0),
  axis.line.x = element_blank(),
  panel.grid.major.x = element_blank(),
  legend.position = "none",
  strip.background = element_rect(fill="white"),
  strip.text = element_text(color = "black", size = 7),
  plot.title = element_text(size=12, hjust=0.5, face="bold", colour="black", vjust=-1),
  plot.subtitle = element_text(size=12, hjust=0.5, color="black")
)

```

What do people in other countries think of Trump's proposed policies

Net approval rating for Trump's proposed policy to...



Plot 2

```

library(ggplot2)
library(dplyr)
library(magrittr)

```

```

library(tidyr)
library(purrr)
library(ggjoy)
library(ggthemes)
library(scales)
library(MASS)
library(grid)

# hwk 2
# recreate 2nd plot
# ignore all polls button, for extra credit you can add it

# graph the upper and lower trend line
# add the *ribbon*
# add plots on top
# indicate 50% mark and make sure axis are same
# get text on the right added

topline <- read.csv('approval_topline.csv')
poll <- read.csv('approval_polllist.csv')

head(topline)

##      president  subgroup modeldate approve_estimate approve_hi approve_lo
## 1 Donald Trump   Voters 3/20/2018      42.01020    46.33399    37.68641
## 2 Donald Trump   Adults 3/20/2018      39.91188    43.94455    35.87922
## 3 Donald Trump All polls 3/20/2018      40.73125    45.33642    36.12609
## 4 Donald Trump   Voters 3/19/2018      41.74577    46.24588    37.24566
## 5 Donald Trump   Adults 3/19/2018      39.92498    44.04139    35.80856
## 6 Donald Trump All polls 3/19/2018      40.73995    45.40366    36.07623
## disapprove_estimate disapprove_hi disapprove_lo      timestamp
## 1          52.84654      56.98879      48.70428 16:10:49 20 Mar 2018
## 2          54.14520      58.21628      50.07412 16:10:40 20 Mar 2018
## 3          53.52146      57.67319      49.36973 16:10:30 20 Mar 2018
## 4          53.18254      57.60318      48.76191 18:48:46 19 Mar 2018
## 5          54.20846      58.39294      50.02398 18:48:37 19 Mar 2018
## 6          53.55711      57.79469      49.31954 18:48:27 19 Mar 2018

top <- topline %>%
  filter(
    subgroup == 'All polls'
  )

df <- poll %>%
  filter(
    subgroup == 'All polls'
  )

df$date <- format(as.Date(strptime(df$timestamp, '%H:%M:%S %d %b %Y')), '%d %b %Y')
df$date <- as.Date(df$date, '%d %b %Y')

df$date <- as.Date(df$startdate, '%m/%d/%Y')

```

```

top$date <- format(as.Date(strptime(top$timestamp, '%H:%M:%S %d %b %Y')), '%d %b %Y')

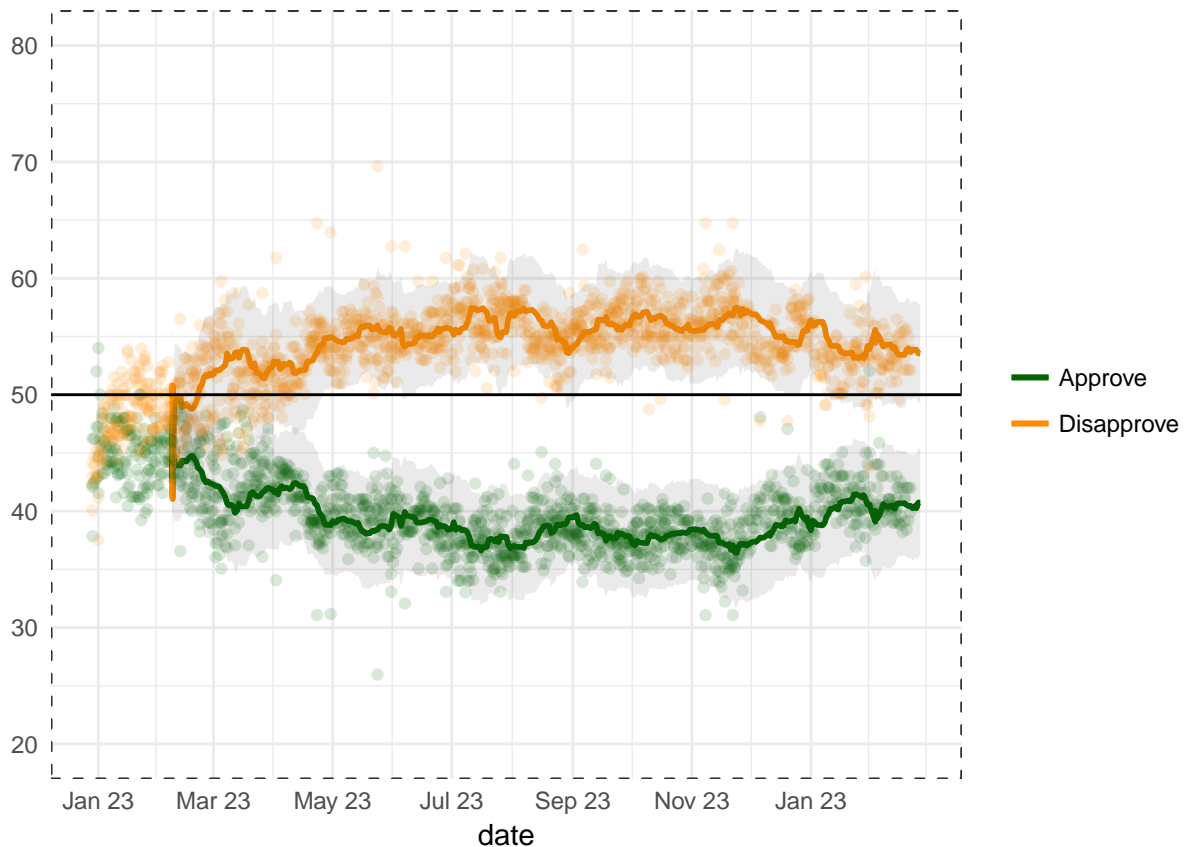
top$date <- as.Date(top$date, '%d %b %Y')

min <- as.Date('2017-1-23')
max <- as.Date('2018-1-23')

top$approve = top$approve_estimate
top$disapprove = top$disapprove_estimate

ggplot() +
  geom_line(data=top, aes(x=date, y=approve_estimate, group=subgroup, color='Approve'), size = 1) +
  geom_line(data=top, aes(x=date, y=top$disapprove_estimate, group=subgroup, color='Disapprove'), size=1) +
  geom_point(data=df, aes(x=date, y=df$adjusted_approve), color='dark green', alpha= .15) +
  geom_point(data=df, aes(x=date, y=df$adjusted_disapprove), color='dark orange', alpha=.15) +
  geom_ribbon(data=top, aes(x=date, ymin=top$approve_hi, ymax=top$approve_lo), alpha=0.1, color=NA) +
  geom_ribbon(data=top, aes(x=date, ymin=top$disapprove_lo, ymax=top$disapprove_hi), alpha=0.1, color=NA) +
  scale_y_continuous('', breaks = seq(20, 80, 10), limits = c(20,80)) +
  scale_x_date(breaks = seq(as.Date('2017-1-23'), as.Date('2018-1-23'), by='2 months'), labels = date_format(breaks, '%d %b %Y')) +
  scale_color_manual(values = c('Approve'='dark green','Disapprove'='dark orange')) +
  theme_bw() +
  theme(
    axis.ticks = element_blank(),
    legend.title = element_blank(),
    panel.border = element_rect(linetype = 'dashed', fill=NA)
  ) +
  geom_hline(yintercept = 50)

```



Plot 3

Question 3. from here on till the separation line below, are all from the midterm code, can be replaced with any code that outputs the final merged data

```
# read in data
load('Dyadicdata.RData')
ls()    ### so the data is loaded in as "x", not sure why

## [1] "datalist" "df"      "gg"      "i"      "main"    "max"
## [7] "min"      "poll"     "regions" "top"     "topline" "urls"
## [13] "x"

Dyadata <- x[which(x$ccode1==2),]  ### isolate only to ccode1 == USA

library(foreign)
y <- read.dta('EPR3CountryNewReduced.dta')  ### y is the EPR3 data frame

polity <- read.csv('p4v2016.csv', stringsAsFactors = F)

load('LJI-estimates-20140422.RData')
```

```

## again it is loaded in as x, so i'll rename it again
ljidata <- x

# merge data
library(countrycode)
Dyadata$name <- countrycode(sourcevar=Dyadata$ccode2,origin="cown", destination='country.name')
y$country <- countrycode(sourcevar=y$cowcode,origin="cown", destination='country.name')
#### according to the code book, the country code was in the "cow" format

#### create the match variable, country_year
Dyadata$cyear <- paste( Dyadata$name, Dyadata$year, sep='_')
y$cyear <- paste( y$country, y$year, sep='_')

# merge in from y
for (i in c("lmtnest", 'exclpop', 'ethfrac')){
  Dyadata[,i] <- y[,i][ match(Dyadata$cyear, y$cyear )]
}

# merge in from polity
#### first, create binary variables bide moc and biautoc
polity$bide moc <- ifelse(polity$polity2 >= 6, 1, 0 )
polity$biautoc <- ifelse(polity$polity2 <= -6, 1, 0)
polity$country <- countrycode(sourcevar=polity$country,origin="country.name", destination='country.name')

## Warning in countrycode(sourcevar = polity$country, origin = "country.name", : Some values were not matched unambiguously
## Warning in countrycode(sourcevar = polity$country, origin = "country.name", : Some strings were matched in multiple places

polity$cyear <- paste( polity$country, polity$year, sep='_')
Dyadata[, "bide moc"] <- polity[, 'bide moc'][ match(Dyadata$cyear, polity$cyear )]
Dyadata[, "biautoc"] <- polity[, 'biautoc'][ match(Dyadata$cyear, polity$cyear )]

# merge in from ljidata
ljidata$X.country. <- countrycode(ljidata$X.ccode., "cown", "country.name")

## Warning in countrycode(ljidata$X.ccode., "cown", "country.name"): Some values were not matched unambiguously

ljidata$cyear <- paste(ljidata$X.country., ljidata$X.year., sep = "_")
Dyadata[, "lji"] <- ljidata[, "X.LJI."][ match(Dyadata$cyear, polity$cyear )]
merged_data <- Dyadata[,c("ccode2", "year", "absidealdiff", "cyear", "lmtnest", "exclpop", "ethfrac", "bide moc", "biautoc", "lji")]
head(merged_data)

##      ccode2 year absidealdiff          cyear lmtnest exclpop
## 1      20 1946      0.313      Canada_1946 2.797281  0.020
## 2      40 1946      0.777      Cuba_1946 1.694107  0.330
## 3      41 1946      1.494      Haiti_1946 2.797281  0.000
## 4      42 1946      0.338 Dominican Republic_1946 2.856470  0.070
## 5      70 1946      1.151      Mexico_1946 3.471967  0.115
## 6      90 1946      1.527      Guatemala_1946 3.763523  0.392
##      ethfrac bide moc biautoc    lji
## 1 0.75499403      1      0 0.8631
## 2 0.03572363      0      0 0.2609
## 3 0.01359123      0      0 0.4888
## 4 0.03698879      0      1 0.3711
## 5 0.30510819      0      1 0.9846
## 6 0.64368415      0      0 0.8780

```

```
# prepare data
for (i in c("lmtnest", "exclpop", "ethfrac", "bidemoc", "biautoc", "lji")){
  merged_data <- merged_data[which(!is.na(merged_data[,i])),]
}
nrow(merged_data) ### rows has no missing value

## [1] 4164
```

up till here are all from mid-term code. can be replaced with any code that outputs the final merged data

=====

now start to test test the uncertainty of the parameters, i choose model 2

```
model_2 <- "absidealdiff ~ lmtnest + exclpop + ethfrac"
lmoutput <- lm( model_2, merged_data)

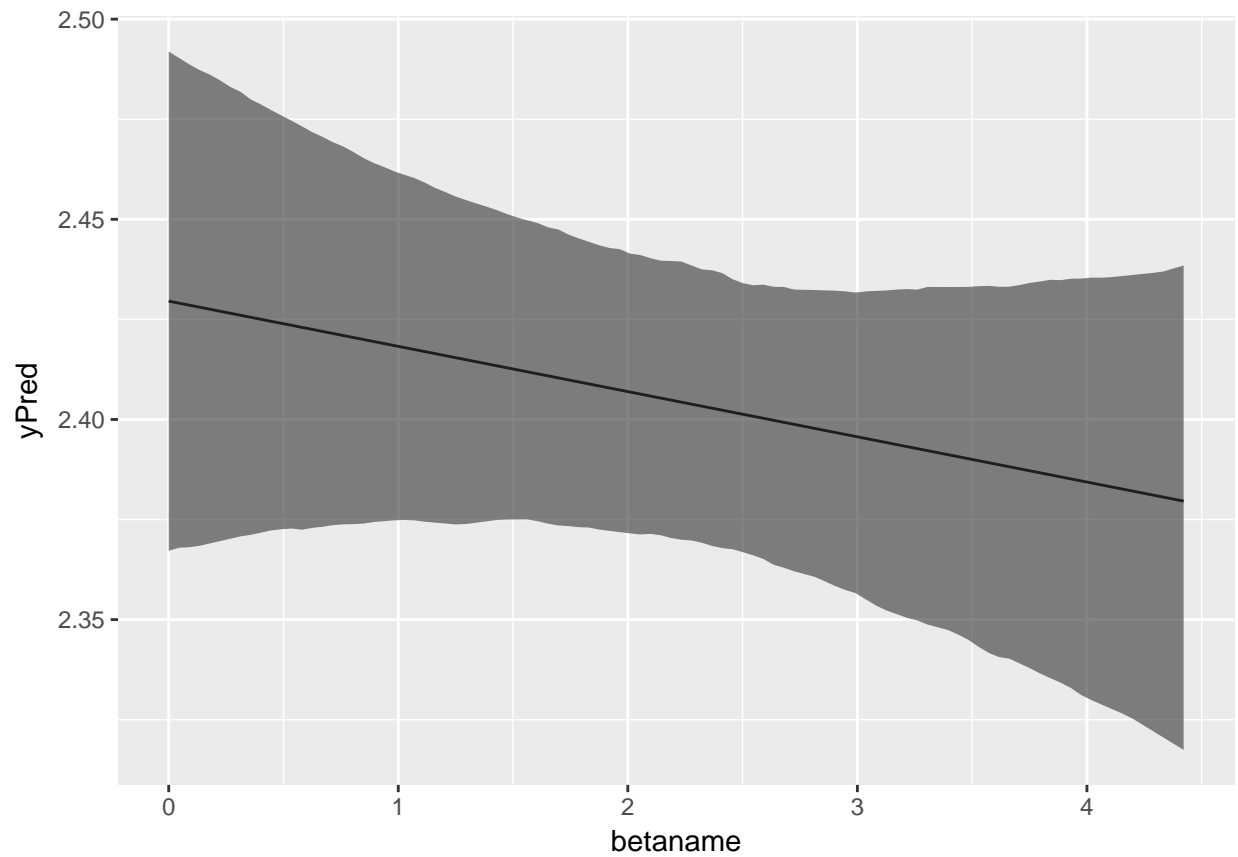
betaMean = coef(lmoutput)
betaDist = vcov(lmoutput)
betaDraws = mvrnorm(1000, betaMean, betaDist)
```

create a function called substantive that takes the beta name and outputs the uncertainty graph

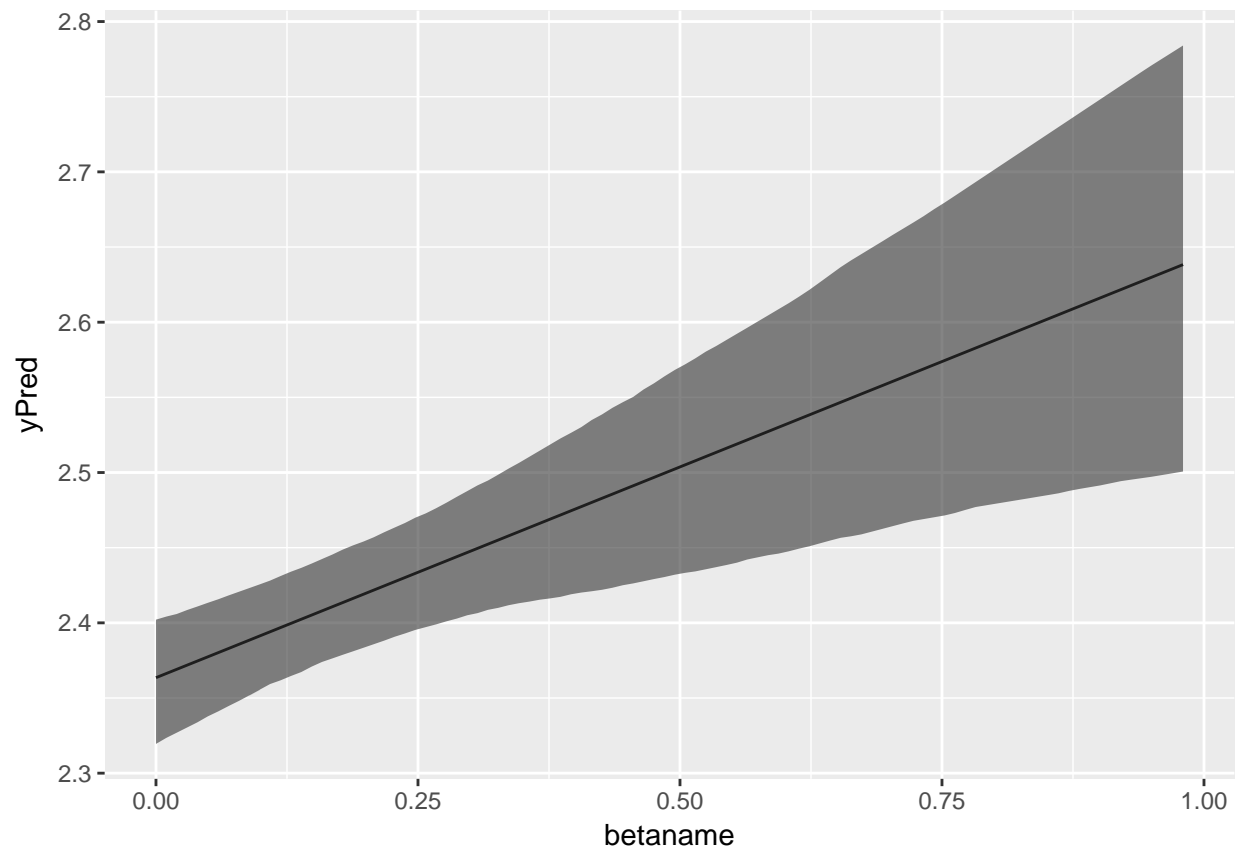
```
substantive <- function(betaname) {
  min <- min(merged_data[,betaname])
  max <- max(merged_data[,betaname])
  Values = seq(min, max, length.out=100)
  scenario = cbind(intercept=rep(1,100), lmtnest=mean(merged_data$lmtnest), exclpop = mean(merged_data$exclpop), ethfrac = mean(merged_data$ethfrac))
  scenario[,betaname] <- Values
  yPred = scenario %*% betaMean
  yPredUncert = scenario %*% t(betaDraws)
  yPredInt = apply(yPredUncert, 1, function(x){quantile(x, c(0.025, 0.975), na.rm=TRUE) })
  yPredInt <- t(yPredInt)
  simAnalysis = data.frame(betaname=Values, yPred=yPred, yPredInt)
  names(simAnalysis)[3:4] = c('q95lo', 'q95hi')

  ggplot(simAnalysis, aes(x=betaname, y=yPred)) + geom_line() + geom_ribbon(aes(ymin=q95lo, ymax=q95hi))
}

substantive('lmtnest')
```

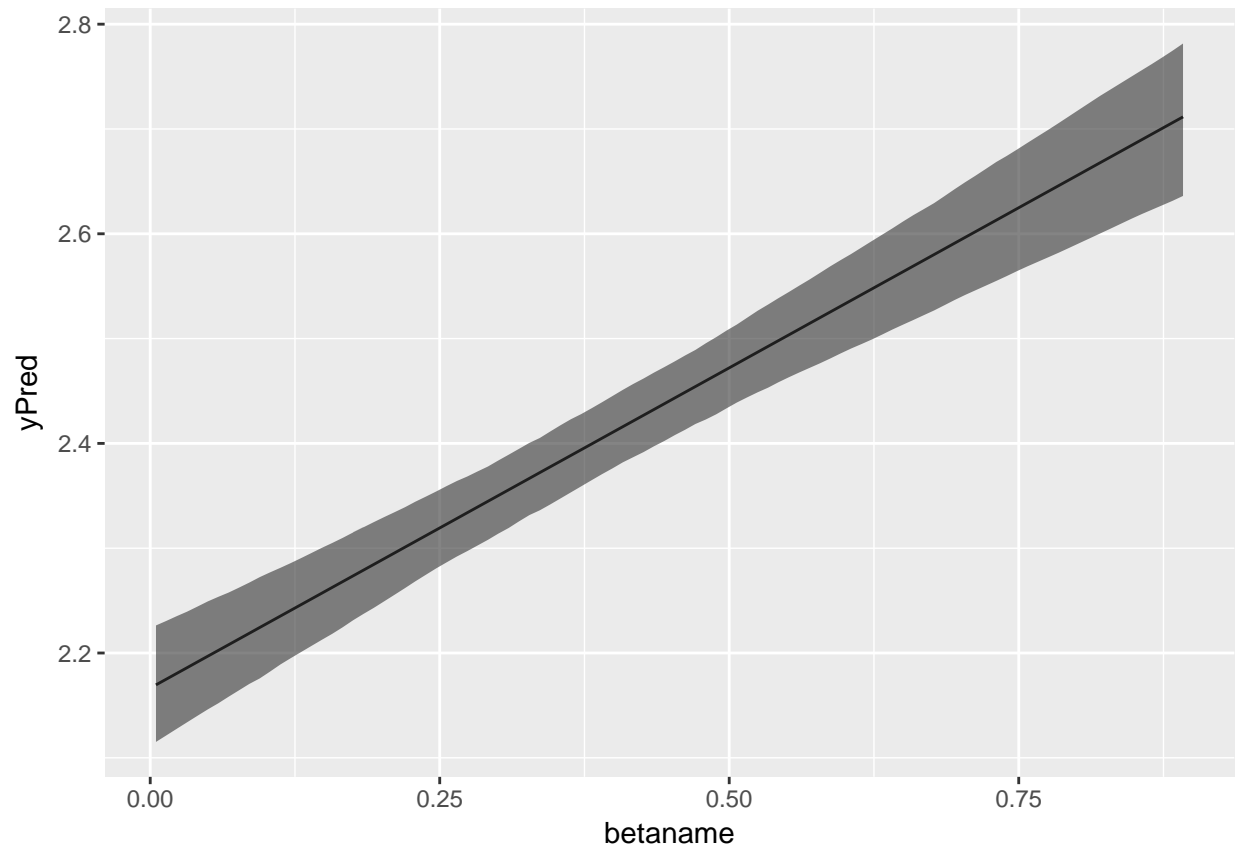


```
# so this beta estimate is not a good one because the ribbon is too wide and the linear relation (beta)  
substantive('exclpop')
```

this is better than the first one. although the uncertainty towards the end is getting bigger. in the

```
substantive('ethfrac')
```



this is the best estimate. the ribbon is narrow so the uncertainty is low.

R Markdown