Research on Youth Suicide and Sexual Orientation is Impacted by High Rates of Missingness

07 January, 2023

Contents

Summary

The present analysis is concerned with very low rates of successful assessment of sexuality (straight, gay, lesbian, bisexual) among youth suspected of suicide. The analysis uses the National Violent Death Reporting System (NVDRS), a surveillance system run by the Centers for Disease Control that covers 43 states, Puerto Rico, and the District of Columbia.

Less than 10% of people suspected of suicide were coded for sexual orientation. This rate of missingness has the potential to strongly bias research on sexuality and suicide mortality, particularly if certain demographic groups are more or less likely to be coded. In this R markdown, we analyze the extent to which coding for sexuality is associated with race, sex, age, and location (state) of death, and the extent to which such biases may impact estimated rates of sexual minority status among suicide victims.

Data preparation and descriptive analyses

We start by loading some packages that will aid analysis.

```
wants <- c("tidyverse","lmerTest","merTools",
           "tidycensus","knitr","usmap", "rstanarm")
has   <- wants %in% rownames(installed.packages())
if(any(!has)) install.packages(wants[!has])
sapply(wants, require, character.only = TRUE)
```

And then load data.

```
load("NVDRS_data.rdata")
dat <- NVDRS_data # shorter name
```

In order to determine whether coding for sexuality is associated with important factors that could bias research, we need to create some demographic (sex, race) and geographic (state of death) variables. Age does not require processing. In the following subsections we create these variables and report descriptives.

State

First we use the NVDRS-provided Federal Information Processing Standards (FIPS) codes to create a state variable.

```
fips_codes <- tidycensus::fips_codes
FIPS <- fips_codes %>% dplyr::select(state, state_code, state_name) %>% unique()
code.index <- match(dat$siteid, as.numeric(FIPS$state_code))
dat <- data.frame(dat, state = FIPS[code.index, "state"] ,
                  state_name = FIPS[code.index, "state_name"])
```

We'll create a reusable function to calculate descriptives moving forward.

```
descriptives_table <- function(var,
                               title){
  tab <- data.frame(table(dat[var]))
  kab <- data.frame(tab[,1],
            tab$Freq,
            round(prop.table(tab$Freq),3) * 100)
  names(kab) <- c(" ", "N", "%")
  # censor cells < 10
  censor <- with(kab, which(N < 10))
  kab[censor,"N"] <- "<10"
  kab[censor,"%"] <- "censored"
  kable(kab, caption=title)
}

descriptives_table(var="state_name", title="Jurisdiction")
```

Table 1: Jurisdiction

	N	%
Alabama	125	1.0
Alaska	176	1.5
Arizona	588	4.9
California	415	3.4
Colorado	481	4.0
Connecticut	125	1.0
Delaware	30	0.2
District of Columbia	16	0.1
Georgia	675	5.6
Hawaii	55	0.5
Illinois	317	2.6
Indiana	382	3.2
Iowa	191	1.6
Kansas	286	2.4
Kentucky	321	2.6
Louisiana	128	1.1
Maine	52	0.4
Maryland	256	2.1
Massachusetts	245	2.0
Michigan	671	5.5
Minnesota	343	2.8
Missouri	233	1.9
Montana	34	0.3
Nebraska	57	0.5
Nevada	145	1.2
New Hampshire	86	0.7
New Jersey	251	2.1
New Mexico	257	2.1
New York	514	4.2
North Carolina	574	4.7
North Dakota	20	0.2
Ohio	792	6.5
Oklahoma	410	3.4
Oregon	357	2.9
Pennsylvania	470	3.9
Puerto Rico	35	0.3
Rhode Island	40	0.3
South Carolina	167	1.4
Utah	453	3.7
Vermont	51	0.4

	N	%
Virginia	358	3.0
Washington	445	3.7
West Virginia	72	0.6
Wisconsin	402	3.3
Wyoming	16	0.1

Sex

Then we create a sex variable.

```
dat$sex_cat <- recode_factor(dat$sex,
                                    "1" = "Male",
                                    "2" = "Female",
                                    "9" = "Unknown")


descriptives_table(var = "sex_cat", title = "Sex")
```

Table 2: Sex

	N	%
Male	9371	77.3
Female	2746	22.7

Race

Then a race variable.

```
dat$race_cat <- recode_factor(dat$raceethnicity_c,
                                    "1" = "White",
                                    "2" = "Black",
                                    "3" = "AmerInd",
                                    "4" = "API",
                                    "5" = "OtherUnspecified",
                                    "6" = "Multiracial",
                                    "7" = "Hispanic",
                                    "9" = "Unknown")

descriptives_table(var = "race_cat", title = "Race")
```

Table 3: Race

	N	%
White	8019	66.2
Black	1278	10.5
AmerInd	369	3
API	516	4.3
OtherUnspecified	44	0.4
Multiracial	282	2.3
Hispanic	1603	13.2
Unknown	<10	censored

Note that there are very few people with `Unknown` race, which will produce unstable model estimates. So, we combine that category together with the `OtherUnspecified` category.

```
dat$race_cat <- recode_factor(dat$race_cat, `OtherUnspecified` = "Unknown")
descriptives_table(var = "race_cat", title = "Race (recoded)")
```

Table 4: Race (recoded)

	N	%
Unknown	50	0.4
White	8019	66.2
Black	1278	10.5
AmerInd	369	3.0
API	516	4.3
Multiracial	282	2.3
Hispanic	1603	13.2

To make model interpretation easier, we relevel the `race_cat` variable so that "White" is a baseline and other races are compared against that category.

```
dat$race_cat <- factor(dat$race_cat,
                       levels=c("White",
                                "Hispanic",
                                "Black",
                                "API",
                                "AmerInd",
                                "Multiracial",
                                "Unknown"))
```

Age

Ages range from 11 to 21.

```
summary(dat$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11.00   16.00   18.00   17.91   20.00   21.00
```

```
sd(dat$age) # standard deviation
```

```
## [1] 2.544577
```

```
descriptives_table(var="age", title = "Age")
```

Table 5: Age

	N	%
11	122	1.0
12	218	1.8
13	420	3.5
14	674	5.6
15	932	7.7
16	1125	9.3
17	1279	10.6
18	1534	12.7
19	1724	14.2
20	1849	15.3
21	2240	18.5

Sexual Orientation

We recode the pre-existing sexual orientation variable according to the NVDRS data dictionary.

```
dat$sexorient_coded <- recode_factor(dat$sexualorientation,
                                     "0" = "Straight",
                                     "1" = "Gay",
                                     "2" = "Lesbian",
                                     "3" = "Bisexual",
                                     "4" = "Unspecified Sexual Minority",
                                     "9" = "Unknown")
# recode purely missing data as Unknown
na.index <- which(is.na(dat$sexorient_coded))
dat[na.index, "sexorient_coded"] <- "Unknown"
```

```
descriptives_table(var="sexorient_coded", title="Sexual orientation")
```

Table 6: Sexual orientation

	N	%
Straight	972	8
Gay	116	1
Lesbian	64	0.5
Bisexual	40	0.3
Unspecified Sexual Minority	<10	censored
Unknown	10917	90.1

Then we create a `sexual_minority` variable that's coded `1` if the person is a sexual minority or `0` if straight, else missing.

```
sexual_minority_strings <- c("Gay", "Lesbian", "Bisexual", "Unspecified Sexual Minority")
dat <- mutate(dat, sexual_minority = as.factor(
                      ifelse(sexorient_coded %in% sexual_minority_strings, 1,
                                  ifelse(sexorient_coded == "Straight", 0,
                                         NA))))

summary(dat$sexual_minority)
```

```
##     0     1  NA's
##   972   228 10917
```

The number of `1` values in the `sexual_minority` variable should match the number of sexual minorities in the `sexorient_coded` variable. Let's double check this.

```
with(dat,
     sum( sexual_minority == 1, na.rm=T ) ==
     sum( sexorient_coded %in% sexual_minority_strings, na.rm=T ))
```

```
## [1] TRUE
```

Next, we create `sexuality_coded` variables which are a `1` if the person's sexuality is coded and a `0` if the status is unknown/uncoded.

Only about 10% of the sample was coded for sexual orientation.

```
dat <- mutate(dat, sexuality_coded_numeric =
                      ifelse(sexualorientation %in% 0:4, 1, 0),
                   sexuality_coded_factor =
                      factor(sexuality_coded_numeric,
```

```
                        levels=0:1, labels=c("no", "yes"))
                )

descriptives_table(var="sexuality_coded_factor", title = "Coded for sexual orientation")
```

Table 7: Coded for sexual orientation

	N	%
no	10917	90.1
yes	1200	9.9

A number of people in the dataset are marked as transgender.

```
summary(as.factor(dat$transgender))
```

```
##     0     1
## 12020    97
```

However, none of these codes are "missing", so we will restrict our analyses to sexual orientation.

```
with(dat, sum(is.na(transgender)))
```

```
## [1] 0
```

Raw rates of successful assessment of sexuality, by group

Now let's look at the raw rates of coding for sexuality broken down by discrete demographic groups.

We'll create a function for this again.

```
orientation_coding_table <- function(var,
                                     title){
  tab <- table(dat[,var], dat[,"sexuality_coded_factor"])
  var_no <- tab[,"no"]; var_yes <- tab[,"yes"]
  var_total <- var_yes + var_no
  percent_yes <- paste0(
      round(
      var_yes/var_total * 100,
      1),
    "%")
  tab <- data.frame(Total.N = var_total,
                        percent.coded = percent_yes)
  colnames(tab) <- c("Total N", "Percent (of N) coded") # replace . with white space
  censor <- with(tab, which(`Total N` < 10))
```

```
  tab[censor,"Total N"] <- "censored"
  kable(tab, caption=title)
}
```

And use the function to produce counts of sexuality codes by demographic.

```
orientation_coding_table(var = "state",
                         title="Coded for sexuality, by state")
```

Table 8: Coded for sexuality, by state

	Total N	Percent (of N) coded
AK	176	16.5%
AL	125	17.6%
AZ	588	6.1%
CA	415	0.5%
CO	481	42.8%
CT	125	19.2%
DC	16	6.2%
DE	30	3.3%
GA	675	8.4%
HI	55	52.7%
IA	191	6.8%
IL	317	1.6%
IN	382	1%
KS	286	4.5%
KY	321	4.7%
LA	128	0.8%
MA	245	28.6%
MD	256	1.6%
ME	52	1.9%
MI	671	1.5%
MN	343	1.5%
MO	233	0.9%
MT	34	35.3%
NC	574	2.1%
ND	20	50%
NE	57	10.5%
NH	86	37.2%
NJ	251	10.8%
NM	257	6.6%
NV	145	17.9%
NY	514	0.8%

	Total N	Percent (of N) coded
OH	792	9.3%
OK	410	26.3%
OR	357	0.6%
PA	470	1.5%
PR	35	31.4%
RI	40	5%
SC	167	10.8%
UT	453	8.4%
VA	358	2.8%
VT	51	2%
WA	445	1.8%
WI	402	55.7%
WV	72	1.4%
WY	16	0%

```
orientation_coding_table(var = "sex_cat",
                         title="Coded for sexuality, by sex")
```

Table 9: Coded for sexuality, by sex

	Total N	Percent (of N) coded
Male	9371	9.3%
Female	2746	12%

```
orientation_coding_table(var = "age",
                         title="Coded for sexuality, by age")
```

Table 10: Coded for sexuality, by age

	Total N	Percent (of N) coded
11	122	6.6%
12	218	6.9%
13	420	6.9%
14	674	8.5%
15	932	10.2%
16	1125	10%
17	1279	10.3%
18	1534	10.4%
19	1724	9.5%
20	1849	10.6%

	Total N	Percent (of N) coded
21	2240	10.4%

```
orientation_coding_table(var="race_cat",
                         title="Coded for sexuality, by race")
```

Table 11: Coded for sexuality, by race

	Total N	Percent (of N) coded
White	8019	10.1%
Hispanic	1603	11.8%
Black	1278	6.1%
API	516	8.1%
AmerInd	369	13%
Multiracial	282	11.7%
Unknown	50	4%

Modeling likelihood of successful assessment of sexual orientation

We use a multilevel logistic regression to simultaneously estimate the relationship between successful coding for sexuality and the deceased person's age, sex, race, and the state where they died.

```
fit <- glmer(sexuality_coded_numeric ~
               age +
               sex_cat +
               race_cat +
               (1|state),
             family="binomial",
             data=dat)
```

We find that likelihood of coding for sexuality is strongly dependent upon all of these demographic factors. (Remember that "White" is the reference category for each of the race coefficients.)

```
summary(fit)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: sexuality_coded_numeric ~ age + sex_cat + race_cat + (1 | state)
##    Data: dat
##
```

```
##      AIC      BIC   logLik deviance df.resid
##   6004.8   6078.9  -2992.4   5984.8    12107
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4021 -0.3004 -0.1560 -0.1102 16.0684
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  state  (Intercept) 2.352    1.534
## Number of obs: 12117, groups:  state, 45
##
## Fixed effects:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -3.82678    0.35183 -10.877  < 2e-16 ***
## age                  0.05717    0.01389   4.117 3.85e-05 ***
## sex_catFemale        0.32153    0.07880   4.080 4.50e-05 ***
## race_catHispanic     0.10590    0.10373   1.021  0.30725
## race_catBlack       -0.38637    0.13362  -2.892  0.00383 **
## race_catAPI         -0.39303    0.19335  -2.033  0.04208 *
## race_catAmerInd     -0.03879    0.19127  -0.203  0.83927
## race_catMultiracial -0.14013    0.22383  -0.626  0.53128
## race_catUnknown     -0.87618    0.75007  -1.168  0.24275
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) age    sx_ctF rc_ctH rc_ctB rc_API rc_cAI rc_ctM
## age         -0.727
## sex_catFeml -0.161  0.147
## rac_ctHspnc -0.061  0.014 -0.018
## race_ctBlck -0.027 -0.014 -0.010  0.119
## race_catAPI -0.016 -0.018 -0.033  0.114  0.073
## rc_ctAmrInd -0.030  0.006 -0.050  0.112  0.055  0.049
## rc_ctMltrcl -0.034  0.016 -0.023  0.093  0.056  0.095  0.097
## rc_ctUnknwn -0.004 -0.001 -0.025  0.028  0.016  0.013  0.031  0.015
```

Exploring the model results

In this section we explore the above model results in detail.

The following code extracts all the fixed effect estimates from the model and expresses them as odds ratios.

```
fe <- fixef(fit)
CI <- confint(fit, parm=names(fe), method="Wald")
```

```
coef.names <- c("Intercept",
                "Age",
                "Female (Ref. Male)",
                "Race:Hispanic (Ref. White)",
                "Race:Black",
                "Race:API",
                "Race:AmerIndian",
                "Race:Multiracial",
                "Race:Unknown")
estimates <- data.frame(Coef = coef.names,
                        Est = round(exp(fe), 2),
                        lower = round(exp(CI[,1]), 2),
                        upper = round(exp(CI[,2]), 2),
                        p.values = round(
                                        summary(fit)$coefficients[,"Pr(>|z|)"],
                                   3)
                        )
estimates <- estimates %>% subset( Coef != "Intercept")
rownames(estimates) <- NULL
kable(estimates)
```

Coef	Est	lower	upper	p.values
Age	1.06	1.03	1.09	0.000
Female (Ref. Male)	1.38	1.18	1.61	0.000
Race:Hispanic (Ref. White)	1.11	0.91	1.36	0.307
Race:Black	0.68	0.52	0.88	0.004
Race:API	0.68	0.46	0.99	0.042
Race:AmerIndian	0.96	0.66	1.40	0.839
Race:Multiracial	0.87	0.56	1.35	0.531
Race:Unknown	0.42	0.10	1.81	0.243

Each year of age is associated with a 6% increased likelihood of coding for sexuality. Women are 38% more likely to be coded for sexual orientation than men. Black people and Asian Pacific Islanders are 32% less likely to be coded for sexuality than White people.

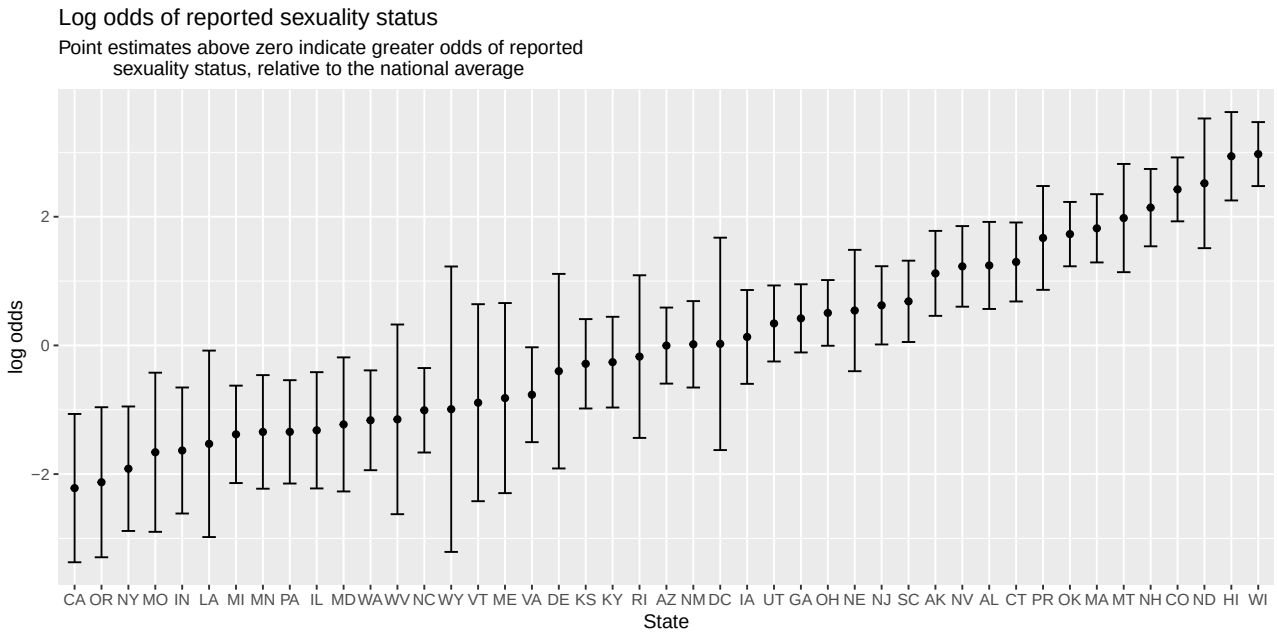
Next, we calculate the likelihood of a person being coded for sexual orientation in each state, after controlling for their race, age, and sex. To do this, we extract the model coefficient estimates for each state as well as the standard errors for those estimates.

```
re <- merTools::REsim(fit, n.sims = 500)
# calculate upper and lower 95%CI for each coef
re$upper <- with( re, mean + sd * qnorm(0.975) )
```

```
re$lower <- with( re, mean - sd * qnorm(0.975) )
raneffs <- with(re, data.frame(State = groupID,
                               Est = mean,
                               lower = lower,
                               upper= upper))
```

We graph the model estimates for state on the log scale.

```
raneffs %>%
  mutate(State = fct_reorder(State, Est)) %>%
  ggplot() +
  aes(x = State, y = Est, ymin=lower, ymax=upper) +
  geom_errorbar(width=0.5) +
  geom_point() +
  ylab("log odds") +
  ggtitle("Log odds of reported sexuality status",
          subtitle="Point estimates above zero indicate greater odds of reported
          sexuality status, relative to the national average")
```



And here's a map giving adjusted odds ratios (i.e., with state coefficients exponentiated).

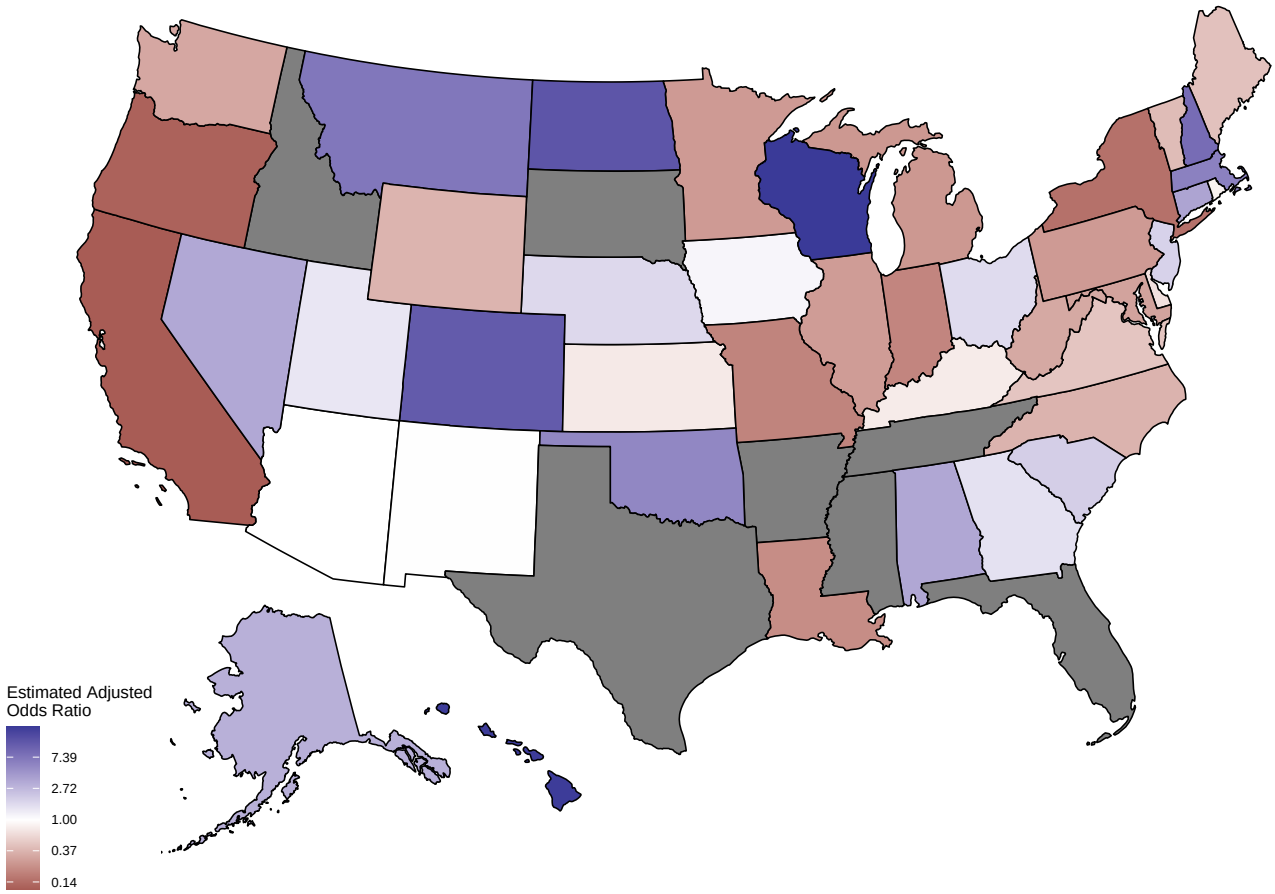
```
usmap_df <- with(raneffs, data.frame(state = State, values = Est))

plot_usmap(regions = "states", data = usmap_df) +
  labs(title = "Adjusted odds of coding for sexual orientation, by state",
       subtitle = "A value of one (1) corresponds to the average of the dataset, after covarying
```

```
for age, sex, and race.\nStates that do not participate in NVDRS are colored grey.") +
  theme(plot.title = element_text(size=20),
        plot.subtitle = element_text(size=13)) +
  scale_fill_gradient2(breaks = -3:3,
                       labels= format(round(exp(-3:3),2), nsmall=2),
                       name = "Estimated Adjusted\nOdds Ratio")
```

Adjusted odds of coding for sexual orientation, by state

A value of one (1) corresponds to the average of the dataset, after covarying for age, sex, and race.
States that do not participate in NVDRS are colored grey.



Accounting for missing sexual orientation information

If we assume that sexuality information is missing completely at random, then our best guess for the "true" rate of sexual minority status among youth suicide decedents is just the raw rate after removing missing codes.

That rate is 19%.

```
(raw_avg <- with(dat, mean(sexual_minority == 1, na.rm=T)))
```

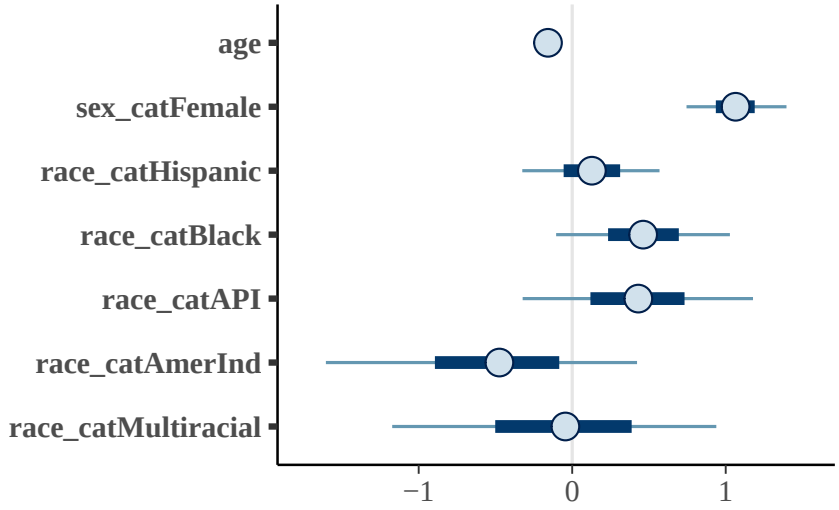
```
## [1] 0.19
```

However, now let's look at patterns of known sexual minority status among different demographic groups, among people for whom we have data.

```
# We use stan to model this pattern so we can get a more accurate
# sense of the uncertainty of the final estimate
sm_model <- stan_glmer( as.numeric(sexual_minority == 1) ~
                age +
                sex_cat +
                race_cat +
                (1|state),
              family="binomial",
              data=dat, cores=4,
              refresh=0)
```

Among people with known sexual orientation, older youths are less likely to identify with a sexual minority status, and women are more than twice as likely to identify as sexual minorities.

```
fixef_pars <- names(fixef(sm_model))
# drop Intercept and Unknown race which is poorly estimated
fixef_pars <- fixef_pars[2:(length(fixef_pars) - 1 )]
#plot log odds. log odds of 1 is about exp(1) = 2.72 greater odds
plot(sm_model, pars=fixef_pars)
```



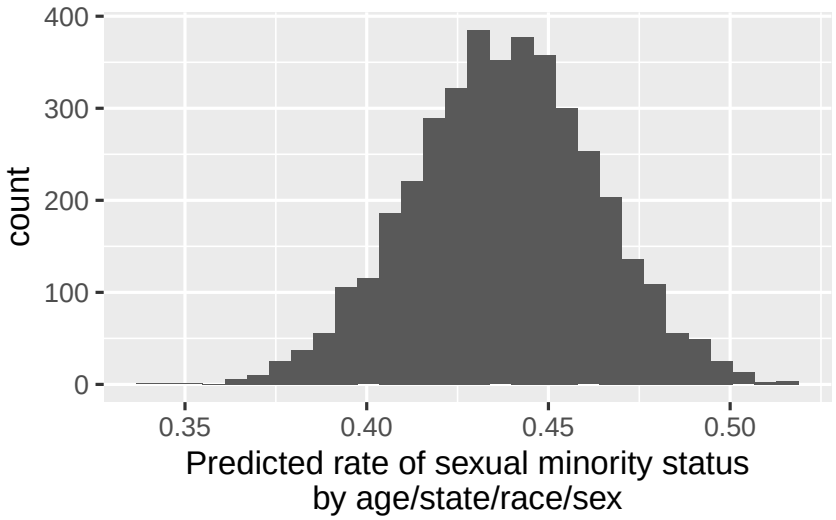
Let's assume that the data are missing at random, i.e., that patterns of sexual orientation by age, race, sex, and state are similar

between people who were and people who were not coded for sexuality. Under this (almost certainly incorrect) assumption, we can use these variables to try to predict overall rates of sexual minority status in the full sample of people who completed suicide.

Here is the model-predicted rate of sexual minority status in the full dataset conditional upon race, age, sex, and state.

```
post <-
  posterior_predict(sm_model,
                    newdata=
                      with(dat,
                           data.frame(age, sex_cat, race_cat, state)),
                    allow.new.levels=TRUE,
                    transform=TRUE)
# rate of sexual minority status for each of 4000 potential datasets
probs <- apply(post, 1, mean)

data.frame(x = probs) %>% ggplot() +
  aes(x=x) +
  geom_histogram(bins=30) +
  xlab("Predicted rate of sexual minority status\nby age/state/race/sex")
```

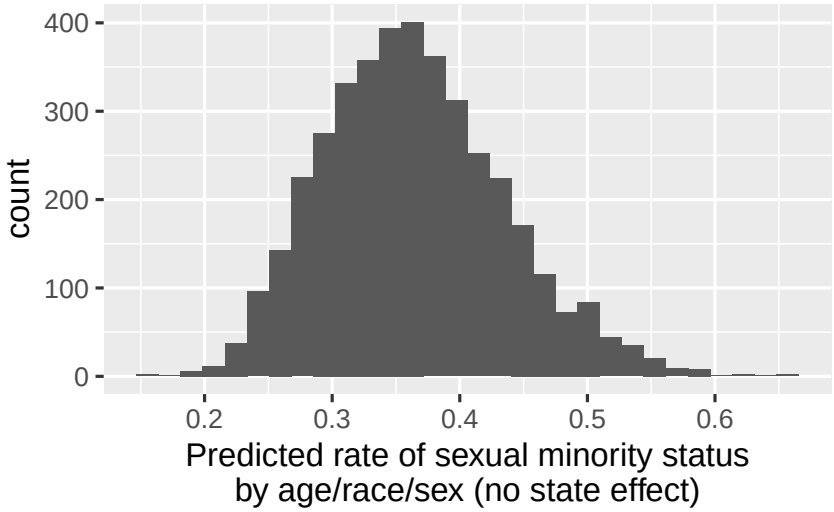


And here is the estimate if we drop state as a predictor:

```
post_no_state <-
  posterior_predict(sm_model,
                    newdata=
                      with(dat,
                           data.frame(age, sex_cat, race_cat, state)),
                    allow.new.levels=TRUE,
                    re.form=NA, # no random effect (i.e., no state effect)
```

```
                    transform=TRUE)
# rate of sexual minorities for each of 4000 potential datasets
probs_no_state <- apply(post_no_state, 1, mean)

data.frame(x = probs_no_state) %>% ggplot() +
  aes(x=x) +
  geom_histogram(bins=30) +
  xlab("Predicted rate of sexual minority status\nby age/race/sex (no state effect)")
```



The model predicts an approximate 44% rate of sexual minority status if we had complete data on sexuality.

```
mean(probs)
```

```
## [1] 0.4377004
```

Stan allows us to calculate posterior predictive intervals, which give a better idea of uncertainty than confidence/credible intervals.

```
quantile(probs, p = c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.3868924 0.4875794
```

In other words, if sexual orientation is missing at random, then the actual rate of sexual minority status is probably more than twice the raw rate (2.3; 95%CI = 2.03-2.56).

```
quantile(
         probs / raw_avg,
         p = c(0.025, # lower CI
               0.5, # point estimate
               0.975) # upper CI
         )
```

```
##     2.5%      50%    97.5%
## 2.036276 2.304288 2.566208
```