IPV Perpetration according to race and employment status

An intersectional analysis using FFCWS data

First, we load the data and the necessary packages:

# packages ----------------------------------------------------------------  
packages <- c("tidyverse", "here", "marginaleffects", "survey", "srvyr", "modelsummary")  
groundhog\_day <- "2024-01-11"  
  
# (install and) load package versions available on the specified day to try  
# to ensure reproducibility  
  
library(groundhog)  
  
groundhog::meta.groundhog(groundhog\_day)  
  
groundhog::groundhog.library(pkg = packages, date = groundhog\_day)  
  
  
dat <- readRDS(file = here::here("01\_data-processing", "data\_private", "data\_final\_imputed\_cases.RDS"))  
  
dat\_weights <- dat |>   
 dplyr::filter(f\_national\_sample == 1) |>   
 srvyr::as\_survey\_rep(  
 repweights = dplyr::contains("f1natwt\_rep"),  
 weights = f1natwt,  
 combined\_weights = TRUE,  
 # why: https://stats.stackexchange.com/questions/409463/duplicating-stata-survey-design-using-svrepdesign-from-survey-package-in-r  
 type = "JKn",  
 scales = 1,  
 rscales = 1,  
 mse = TRUE  
 )

We create 6 models - three weighted, three unweighted:

m1.simple <- lm(  
 ipv\_prop ~ f\_age,  
 data = dat  
)  
  
m1.weighted <- survey::svyglm(  
 ipv\_prop ~ f\_age,  
 design = dat\_weights  
)  
  
m2.simple <- lm(  
 ipv\_prop ~ f\_age + f\_race,  
 data = dat  
)  
  
m2.weighted <- survey::svyglm(  
 ipv\_prop ~ f\_age + f\_race,  
 design = dat\_weights  
)  
  
m3.simple <- lm(  
 ipv\_prop ~ f\_age + f\_race + f\_employment,  
 data = dat  
)  
  
m3.weighted <- survey::svyglm(  
 ipv\_prop ~ f\_age + f\_race + f\_employment,  
 design = dat\_weights  
)  
  
all.r2 <- list(m1.simple, m1.weighted, m2.simple, m2.weighted, m3.simple, m3.weighted) |>   
 purrr::map(  
 .f = ~ modelsummary::get\_gof(.x) |> as.tibble() |> select(dplyr::contains("r.squared"))  
 ) |> list\_rbind()  
  
all.r2 |> mutate(  
 model = c("m1.simple", "m1.weighted", "m2.simple", "m2.weighted", "m3.simple", "m3.weighted")  
) |> dplyr::relocate(model)

# A tibble: 6 × 3  
 model r.squared adj.r.squared  
 <chr> <dbl> <dbl>  
1 m1.simple 0.00389 0.00326  
2 m1.weighted 0.00277 -50.0   
3 m2.simple 0.0129 0.0116   
4 m2.weighted 0.0122 -51.2   
5 m3.simple 0.0152 0.0133   
6 m3.weighted 0.0124 -52.9

# for our survey models, we want modelsummary() to retrieve the AIC value that is stored in the model object  
# instead of trying to calculate it as there's some sort of bug there (still to ask on StackOverflow about this)  
AIC.svrepglm <- function(model){  
 AIC <- model$aic  
   
 list(AIC = AIC)  
}  
  
  
modelsummary::modelsummary(  
 models = list(  
 "A" = m1.simple,   
 "AR" = m2.simple,   
 "ARE" = m3.simple,   
 "A" = m1.weighted,   
 "AR" = m2.weighted,   
 "ARE" = m3.weighted  
 ), stars = TRUE, output = "gt", fmt = fmt\_statistic("rmse" = 5)) |>   
 gt::tab\_spanner(label = "Unweighted", columns = 2:4) |>   
 gt::tab\_spanner(label = "Weighted", columns = 5:7)

|  | Unweighted | | | Weighted | | |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | AR | ARE | A | AR | ARE |
| (Intercept) | 0.172\*\*\* | 0.149\*\*\* | 0.166\*\*\* | 0.138\*\* | 0.123\*\* | 0.128\*\* |
|  | (0.016) | (0.017) | (0.019) | (0.041) | (0.043) | (0.045) |
| f\_age | -0.001\* | -0.001\* | -0.001+ | -0.001 | -0.001 | -0.001 |
|  | (0.001) | (0.001) | (0.001) | (0.001) | (0.001) | (0.001) |
| f\_raceBlack |  | 0.030\*\*\* | 0.027\*\*\* |  | 0.027+ | 0.026+ |
|  |  | (0.008) | (0.008) |  | (0.014) | (0.015) |
| f\_employmentEmployed |  |  | -0.021+ |  |  | -0.007 |
|  |  |  | (0.011) |  |  | (0.016) |
| Num.Obs. | 1585 | 1585 | 1585 | 1585 | 1585 | 1585 |
| R2 | 0.004 | 0.013 | 0.015 | 0.003 | 0.012 | 0.012 |
| R2 Adj. | 0.003 | 0.012 | 0.013 | -49.955 | -51.158 | -52.944 |
| AIC | -1346.2 | -1358.5 | -1360.3 | 106.1 | 93.1 | 94.8 |
| BIC | -1330.1 | -1337.1 | -1333.4 | 9226.2 | 9151.8 | 9105.0 |
| Log.Lik. | 676.110 | 683.273 | 685.141 | -4602.064 | -4561.158 | -4534.057 |
| F | 6.187 | 10.302 | 8.123 | 0.571 | 3.577 | 3.607 |
| RMSE | 0.15795 | 0.15723 | 0.15705 | 0.15984 | 0.15838 | 0.15826 |
| + p < 0.1, \* p < 0.05, \*\* p < 0.01, \*\*\* p < 0.001 | | | | | | |