Final Model

Team

2025-12-08

```
# Path to cleaned CFPS person-year dataset
clean_path <- "data/cfps_model_ready.csv"

cfps <- read.csv(clean_path, stringsAsFactors = FALSE)

# Basic info

cat("Number of rows:", nrow(cfps), "\n")
```

```
## Number of rows: 51224
```

```
cat("Number of columns:", ncol(cfps), "\n\n")
```

```
## Number of columns: 18
```

```
# Peek at column names
names(cfps)
```

```
##  [1] "pid"           "cyear"         "cmonth"        "provcd"
##  [5] "countyid"      "cid"           "urban"         "cesd20"
##  [9] "fid"           "subsample"     "subpopulation" "wt_natcs"
## [13] "wt_natpn10"    "age"           "gender"        "education"
## [17] "marital"       "health"
```

```
str(cfps)
```

```
## 'data.frame':    51224 obs. of  18 variables:
##  $ pid          : num  1.0e+08 1.0e+08 1.2e+08 1.3e+08 1.3e+08 ...
##  $ cyear        : int  2022 2022 2022 2022 2022 2022 2022 2022 2022 2022 ...
##  $ cmonth       : int  6 6 7 7 6 7 6 7 7 7 ...
##  $ provcd       : int  11 11 12 13 13 13 13 13 13 13 ...
##  $ countyid     : int  45 45 189 363 48 53 48 48 48 50 ...
##  $ cid          : int  624942 624942 800637 462546 119300 121400 119400 119500 119500 188442
##  $ urban        : int  1 1 1 1 1 0 1 0 0 1 ...
##  $ cesd20       : int  6 5 6 28 20 8 20 6 32 14 ...
##  $ fid          : int  100051 100051 100160 100551 100724 130463 100765 100782 100782 101023
##  $ subsample    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ subpopulation: int  6 6 6 6 6 6 6 6 6 6 ...
##  $ wt_natcs     : num  NA NA 1.16 1.13 1.33 ...
```

```
##  $ wt_natpn10   : num  NA NA 1.44 1.6 NA ...
##  $ age          : int  53 56 31 34 34 58 29 52 48 41 ...
##  $ gender       : int  0 1 0 1 0 1 0 1 0 0 ...
##  $ education    : int  5 5 7 7 6 3 4 4 4 5 ...
##  $ marital      : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ health       : int  3 3 3 3 3 3 5 5 3 2 ...
```

```
# Unique survey years and counts

cat("Unique years (cyear):\n")
```

```
## Unique years (cyear):
```

```
print(sort(unique(cfps$cyear)))
```

```
## [1] 2016 2017 2018 2019 2020 2022
```

```
cat("\nCount by year:\n")
```

```
##
## Count by year:
```

```
print(table(cfps$cyear, useNA = "ifany"))
```

```
##
##  2016  2017  2018  2019  2020  2022
## 11308  1498 12474   332 12806 12806
```

```
cat("\nUnique months (cmonth):\n")
```

```
##
## Unique months (cmonth):
```

```
print(sort(unique(cfps$cmonth)))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12
```

```
cat("\nCross-tab of year × month (first few):\n")
```

```
##
## Cross-tab of year × month (first few):
```

```
tab_year_month <- table(cfps$cyear, cfps$cmonth, useNA = "ifany")
tab_year_month[ , colSums(tab_year_month) > 0, drop = FALSE]
```

```
##
##           1    2    3    4    5    6    7    8    9   10   11   12
##   2016    0    0    0    0    0  110 4983 4834  533  463  197  188
##   2017  284  307  646  261    0    0    0    0    0    0    0    0
##   2018    0    0    0    0    0   96 4599 5815  508  711  526  219
##   2019  167   25   38   39   63    0    0    0    0    0    0    0
##   2020    0    0    0    0    0    0 5036 6135  629  449  366  191
##   2022    0    0    0    0  245  783 4306 3595 1549 1118  885  325
```

```r
# How many unique persons?

n_pid <- length(unique(cfps$pid))
cat("Number of unique persons (pid):", n_pid, "\n\n")
```

```
## Number of unique persons (pid): 12806
```

```r
# Check uniqueness of person-year rows

key_df <- cfps[, c("pid", "cyear")]
dup_key <- duplicated(key_df)

cat("Number of duplicate pid-cyear combinations:", sum(dup_key), "\n")
```

```
## Number of duplicate pid-cyear combinations: 0
```

```r
if (sum(dup_key) > 0) {
cat("Showing first few duplicates:\n")
print(head(cfps[dup_key, c("pid", "cyear")]))
}
```

```r
core_vars <- c(
"pid", "code", "fid", "cid", "countyid", "provcd",
"psu", "subsample", "subpopulation",
"wt_natcs", "wt_natpn10",
"cyear", "cmonth",
"qa001y", "qa001m",
"cesd20", "cesd8",
"age", "gender", "education", "urban",
"marital", "health", "party", "hukou", "ethnicity"
)

core_vars <- intersect(core_vars, names(cfps))

missing_prop_core <- sapply(core_vars, function(v) {
mean(is.na(cfps[[v]]))
})

cat("Missingness for core variables:\n")
```

```
## Missingness for core variables:
```

```r
print(round(missing_prop_core, 3))
```

```
##           pid           fid           cid      countyid        provcd
##         0.000         0.000         0.046         0.002         0.001
##     subsample subpopulation      wt_natcs    wt_natpn10         cyear
##         0.000         0.000         0.062         0.170         0.000
##        cmonth        cesd20           age        gender     education
##         0.000         0.048         0.000         0.000         0.004
```

```
##         urban       marital        health
##         0.000         0.014         0.000
```

```
# CESD-20 summary by year

years <- sort(unique(cfps$cyear))
for (yy in years) {
cat("\nYear:", yy, "\n")
vals <- cfps$cesd20[cfps$cyear == yy]
cat("  N (non-missing):", sum(!is.na(vals)), "\n")
cat("  Mean (sd):",
round(mean(vals, na.rm = TRUE), 2), "(",
round(sd(vals, na.rm = TRUE), 2), ")\n")
}
```

```
##
## Year: 2016
##   N (non-missing): 11308
##   Mean (sd): 12.22 ( 7.93 )
##
## Year: 2017
##   N (non-missing): 1498
##   Mean (sd): 13.15 ( 7.41 )
##
## Year: 2018
##   N (non-missing): 11919
##   Mean (sd): 12.94 ( 7.67 )
##
## Year: 2019
##   N (non-missing): 318
##   Mean (sd): 13.04 ( 7.99 )
##
## Year: 2020
##   N (non-missing): 11978
##   Mean (sd): 13.15 ( 8.1 )
##
## Year: 2022
##   N (non-missing): 11743
##   Mean (sd): 13.8 ( 8.36 )
```

```
# Gender distribution by year (if coded as 0/1 or 1/2, this is still informative)

if ("gender" %in% names(cfps)) {
cat("\nGender distribution by year:\n")
print(table(cfps$cyear, cfps$gender, useNA = "ifany"))
}
```

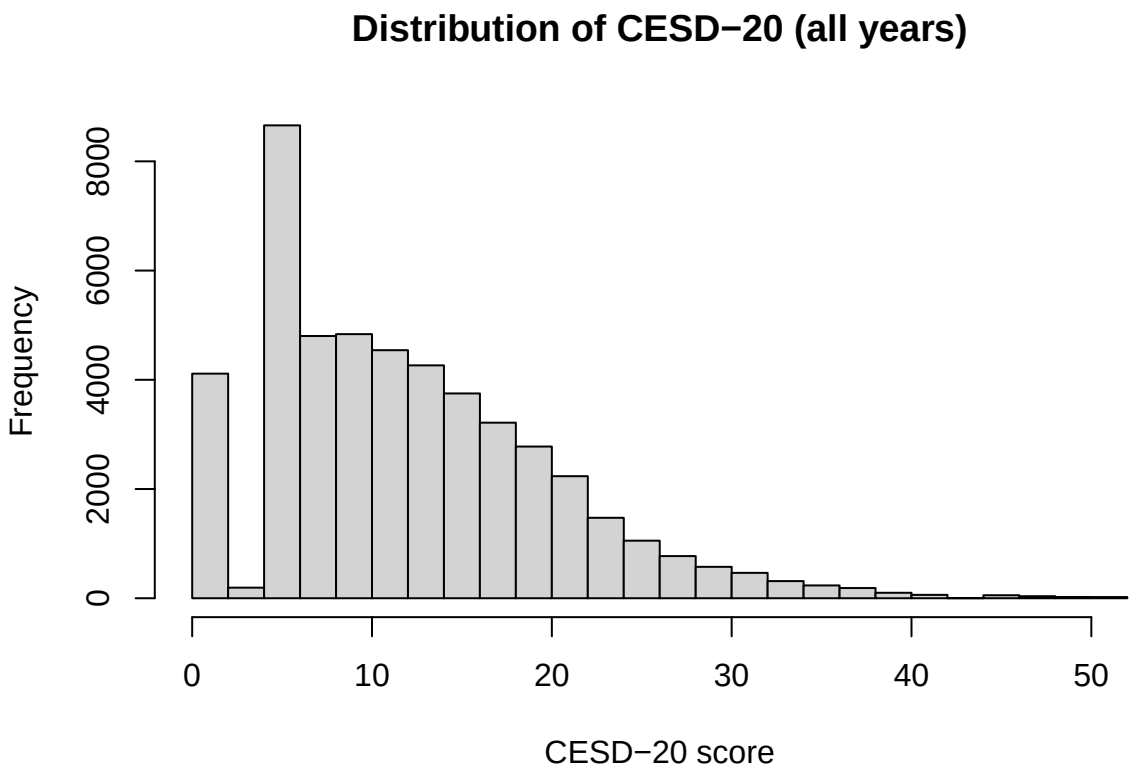
```
##
## Gender distribution by year:
```

```
##
##           0    1
##   2016 5881 5427
##   2017  628  870
##   2018 6367 6107
##   2019  150  182
##   2020 6516 6290
##   2022 6517 6289
```

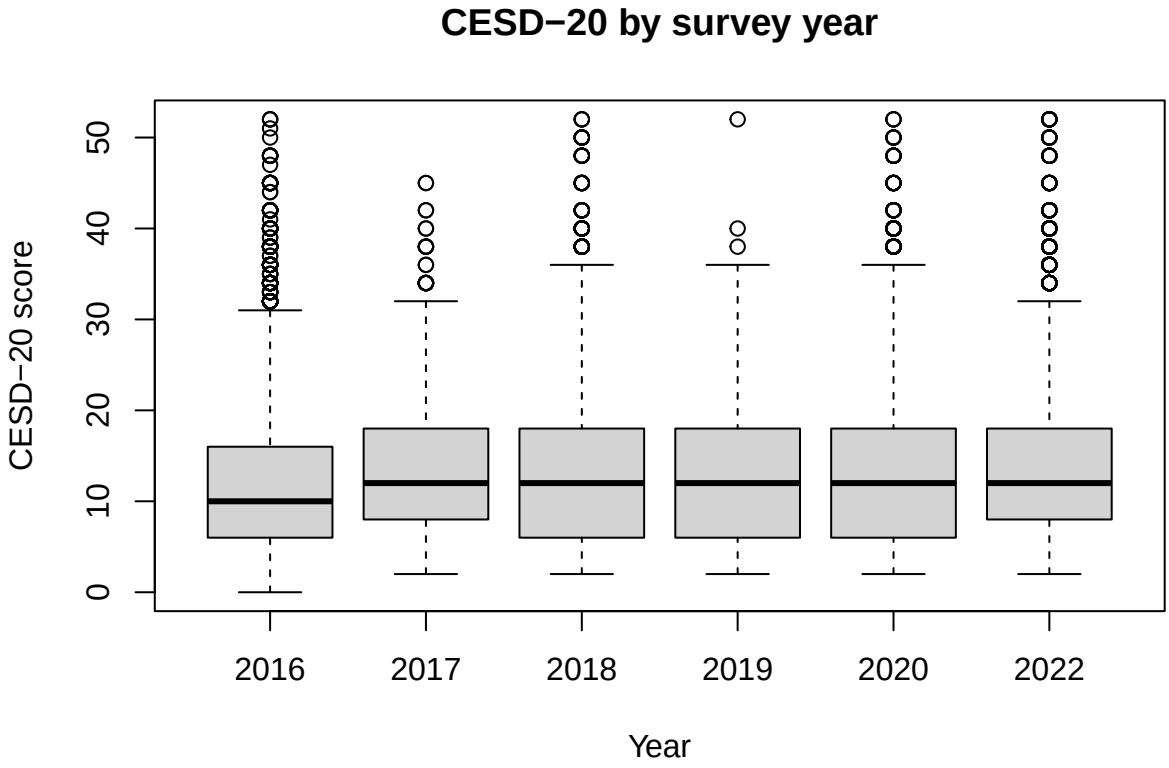
```r
# Urban vs rural by year

if ("urban" %in% names(cfps)) {
cat("\nUrban (0/1) by year:\n")
print(table(cfps$cyear, cfps$urban, useNA = "ifany"))
}
```

```
##
## Urban (0/1) by year:
##
##          -9    0    1
##   2016   46 5941 5321
##   2017   33  620  845
##   2018  375 5914 6185
##   2019    9  169  154
##   2020  690 5958 6158
##   2022   36 6070 6700
```

```r
hist(
cfps$cesd20,
breaks = 30,
main = "Distribution of CESD-20 (all years)",
xlab = "CESD-20 score"
)
```




```
boxplot(
cesd20 ~ cyear,
data = cfps,
main = "CESD-20 by survey year",
xlab = "Year",
ylab = "CESD-20 score"
)
```


CESD−20 by survey year
CESD−20 score
0
10
20
30
40
50
2016
2017
2018
2019
2020
2022
Year