Double lasso replication of main tables

2025-08-24

Packages and libraries required in the following tables.

```r
install.packages(c("tidyverse", "glmnet", "sandwich", "lmtest", "car", "modelsummary", "broom", "haven"
```

```
## Installing packages into '/usr/local/lib/R/site-library'
## (as 'lib' is unspecified)
```

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(glmnet)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-10
```

```r
library(sandwich)
library(lmtest)
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library(car)
```

```
## Loading required package: carData
##
```

```
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following object is masked from 'package:purrr':
##
##     some
```

```r
library(modelsummary)
library(broom)
library(haven)
```

Replication of Table 1 using double lasso.

```r
############################## TABLE 1 ############################################

## DATA AND MACROS ##

# Baseline list of controls
gpcontrols <- c("GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
                "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
                "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st")


# Full list of candidates
full_controls_candidates <- c(
  # baseline elements
  "GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
  "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
  "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st",

  # Added : from the census
  "CENSUS_PCA2001_tot_pop", "CENSUS_PCA2001_tot_lit", "CENSUS_PCA2001_tot_sc",
  "CENSUS_PCA2001_tot_st", "CENSUS_PCA2001_tot_aglb", "CENSUS_PCA2001_tot_nm_hh",
  "CENSUS_PCA2001_tot_cult", "CENSUS_VD2001_power_dom", "CENSUS_VD2001_drnk_wat_f",
  "CENSUS_VD2001_edu_fac", "CENSUS_VD2001_medi_fac"
)

# Dependent variables
incum_dep_vars1 <- c("INC05_running", "INC05_voteshare",
                     "INCSPOUSE05_running", "INCSPOUSE05_voteshare",
                     "INCOTHER05_running", "INCOTHER05_voteshare")

# Data loading and filtering
data <- haven::read_dta("~/work/Electoral data cleaned.dta")

data_filtered <- data %>%
  filter(RES10_gender == 0, SAMPLE_hhsurvey == 1, GP_tag == 1, INC05_can_run == 1) %>%
  mutate(
    FAMnotINC05_running   = INCorFAM05_running  - INC05_running,
    FAMnotINC05_voteshare = INCorFAM05_voteshare - INC05_voteshare,
    FAMnotINC05_won       = INCorFAM05_won - INC05_won
  )

data_filtered %>% select(all_of(full_controls_candidates)) %>% summary()
```

```
##  GP_population       GP_lit           GP_sc              GP_st
##  Min.   : 3250   Min.   :0.3205   Min.   :0.009154   Min.   :0.000000
##  1st Qu.: 4610   1st Qu.:0.5096   1st Qu.:0.150252   1st Qu.:0.000000
##  Median : 5550   Median :0.5599   Median :0.232531   Median :0.001974
##  Mean   : 5762   Mean   :0.5478   Mean   :0.226674   Mean   :0.089329
##  3rd Qu.: 6658   3rd Qu.:0.6011   3rd Qu.:0.301907   3rd Qu.:0.124034
##  Max.   :12274   Max.   :0.6856   Max.   :0.548745   Max.   :0.705038
##  GP_nbvillages     RES00_gender      RES00_obc         RES00_sc
##  Min.   : 1.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 4.000   Median :0.0000   Median :0.0000   Median :0.0000
##  Mean   : 4.493   Mean   :0.2895   Mean   :0.1711   Mean   :0.2566
##  3rd Qu.: 5.000   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000
##  Max.   :12.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##     RES00_st        RES10_obc         RES10_sc          RES10_st
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.0000   Median :0.0000   Median :0.00000   Median :0.00000
##  Mean   :0.1447   Mean   :0.1316   Mean   :0.01316   Mean   :0.05263
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
##    RES05_obc         RES05_sc         RES05_st       CENSUS_PCA2001_tot_pop
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   : 2782
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.: 3950
##  Median :0.0000   Median :0.0000   Median :0.00000   Median : 4711
##  Mean   :0.1184   Mean   :0.2368   Mean   :0.07237   Mean   : 5303
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.: 5735
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :25548
##  CENSUS_PCA2001_tot_lit CENSUS_PCA2001_tot_sc CENSUS_PCA2001_tot_st
##  Min.   :  920          Min.   :   71         Min.   :   0.0
##  1st Qu.: 1801          1st Qu.:  629         1st Qu.:   0.0
##  Median : 2252          Median : 1033         Median :   5.5
##  Mean   : 2500          Mean   : 1225         Mean   : 485.8
##  3rd Qu.: 2877          3rd Qu.: 1579         3rd Qu.: 619.8
##  Max.   :15330          Max.   :11400         Max.   :6915.0
##  CENSUS_PCA2001_tot_aglb CENSUS_PCA2001_tot_nm_hh CENSUS_PCA2001_tot_cult
##  Min.   :  3.0           Min.   : 370.0           Min.   : 438.0
##  1st Qu.: 37.0           1st Qu.: 571.5           1st Qu.: 779.8
##  Median : 67.5           Median : 693.0           Median : 986.5
##  Mean   : 87.7           Mean   : 776.6           Mean   :1062.6
##  3rd Qu.:116.2           3rd Qu.: 832.8           3rd Qu.:1198.5
##  Max.   :351.0           Max.   :3786.0           Max.   :4182.0
##  CENSUS_VD2001_power_dom CENSUS_VD2001_drnk_wat_f CENSUS_VD2001_edu_fac
##  Min.   :0.0000          Min.   :0.5000           Min.   :0.4000
##  1st Qu.:0.0000          1st Qu.:1.0000           1st Qu.:0.8750
##  Median :0.2111          Median :1.0000           Median :1.0000
##  Mean   :0.3223          Mean   :0.9604           Mean   :0.9275
##  3rd Qu.:0.6000          3rd Qu.:1.0000           3rd Qu.:1.0000
##  Max.   :1.0000          Max.   :1.0000           Max.   :1.0000
##  CENSUS_VD2001_medi_fac
##  Min.   :0.0000
##  1st Qu.:0.1917
##  Median :0.4000
##  Mean   :0.4267
##  3rd Qu.:0.6667
##  Max.   :1.0000
```

```r
## FUNCTIONS ##

# Non zero coefficients, no intercept
get_nonzero_names <- function(cf, drop = "(Intercept)") {
  cf_mat <- as.matrix(cf)
  nz <- which(cf_mat != 0)
  vars <- rownames(cf_mat)[nz]
  setdiff(vars, drop)
}


# Double lasso function
double_lasso_fit <- function(data_df, y_name, d_names, x_candidates, baseline_names = NULL, fe_name = "
  # FE
  if (!is.null(fe_name)) {
    fe_vec <- data_df[[fe_name]]
    fe_mat <- model.matrix(~ factor(fe_vec) - 1)
    colnames(fe_mat) <- paste0("fe_", levels(factor(fe_vec)))
  } else {
    fe_mat <- NULL
  }

  # X : candidates
  X_all <- data_df %>% dplyr::select(all_of(x_candidates))
  X_mat <- model.matrix(~ . - 1, data = X_all)

  # Filter NA rows
  keep_rows_X <- complete.cases(X_mat)
  if (!is.null(fe_mat)) {
    keep_rows_FE <- complete.cases(fe_mat)
    keep_rows <- keep_rows_X & keep_rows_FE
    X_mat <- X_mat[keep_rows, , drop = FALSE]
    fe_mat <- fe_mat[keep_rows, , drop = FALSE]
  } else {
    X_mat <- X_mat[keep_rows_X, , drop = FALSE]
    keep_rows <- keep_rows_X
  }

  baseline_idx <- if (!is.null(baseline_names)) {
    which(colnames(X_mat) %in% baseline_names)
  } else integer(0)

  if (!is.null(fe_mat)) {
    X_mat <- cbind(X_mat, fe_mat)
    fe_idx <- (ncol(X_mat) - ncol(fe_mat) + 1):ncol(X_mat)
  } else {
```

```
  fe_idx <- integer(0)
}

# Y and D
missing_d <- setdiff(d_names, names(data_df))
if (length(missing_d) > 0) {
  for (nm in missing_d) {
    parts <- strsplit(nm, ":", fixed = TRUE)[[1]]
    stopifnot(all(parts %in% names(data_df)))
    data_df[[nm]] <- data_df[[parts[1]]] * data_df[[parts[2]]]
  }
}

y_vec <- as.numeric(data_df[[y_name]])
D_mat <- as.matrix(data_df %>% dplyr::select(all_of(d_names)))
if (is.null(ncol(D_mat))) D_mat <- matrix(D_mat, ncol = 1)

# Filter y_vec and D_mat
y_vec <- y_vec[keep_rows]
D_mat <- D_mat[keep_rows, , drop = FALSE]
if (!is.null(se_cluster)) se_cluster <- se_cluster[keep_rows]

# Checks
y_constant <- length(unique(y_vec)) < 2
d_constant <- apply(D_mat, 2, function(z) length(unique(z)) < 2)

# Y ~ D + X
sel_y <- character(0); lambda_y <- NA
if (!y_constant) {
  X_y <- cbind(D_mat, X_mat)
  pf_y <- c(rep(0, ncol(D_mat)), rep(1, ncol(X_mat)))
  if (length(fe_idx) > 0) pf_y[ncol(D_mat) + fe_idx] <- 0

  if (all(pf_y == 0)) {
    sel_y <- character(0)
    lambda_y <- NA
  } else {
    set.seed(123)
    fit1 <- tryCatch({
      cv.glmnet(X_y, y_vec, alpha = 1, family = family, penalty.factor = pf_y)
    }, error = function(e) NULL)

    if (!is.null(fit1)) {
      coef1 <- coef(fit1, s = "lambda.min")
      cf_mat1 <- as.matrix(coef1)
      nz1 <- which(cf_mat1 != 0)
      sel_y <- rownames(cf_mat1)[nz1]
      sel_y <- setdiff(sel_y, c("(Intercept)", colnames(D_mat)))
      lambda_y <- fit1$lambda.min
    } else {
      sel_y <- character(0)
      lambda_y <- NA
    }
  }
}

# D_k ~ X
pf_d <- rep(1, ncol(X_mat))
if (length(baseline_idx) > 0) pf_d[baseline_idx] <- 1
if (length(fe_idx) > 0) pf_d[fe_idx] <- 0

sel_d <- character(0)
d_lambdas <- rep(NA, ncol(D_mat))

if (!all(pf_d == 0)) {
  for (k in seq_len(ncol(D_mat))) {
    if (d_constant[k]) next
    set.seed(123)
    fitk <- tryCatch({
      cv.glmnet(X_mat, D_mat[, k], alpha = 1, family = "gaussian", penalty.factor = pf_d)
    }, error = function(e) NULL)

    if (!is.null(fitk)) {
      coefk <- coef(fitk, s = "lambda.min")
      cf_matk <- as.matrix(coefk)
      nz_k <- which(cf_matk != 0)
      sel_k <- rownames(cf_matk)[nz_k]
      sel_k <- setdiff(sel_k, "(Intercept)")
      sel_d <- union(sel_d, sel_k)
      d_lambdas[k] <- fitk$lambda.min
    }
  }
}

# Union and final regression
selected_controls <- union(sel_y, sel_d)
if (length(fe_idx) > 0) selected_controls <- union(selected_controls, colnames(X_mat)[fe_idx])

X_final <- X_mat[, selected_controls, drop = FALSE]
final_df <- data.frame(Y = y_vec, D_mat, X_final)
colnames(final_df)[2:(1 + ncol(D_mat))] <- colnames(D_mat)

rhs <- paste(c(colnames(D_mat), colnames(X_final)), collapse = " + ")
f_final <- as.formula(paste("Y ~", rhs))
lm_final <- lm(f_final, data = final_df)

vc <- if (!is.null(se_cluster)) {
  sandwich::vcovCL(lm_final, cluster = se_cluster)
} else {
  sandwich::vcovHC(lm_final, type = "HC1")
}

coefs_rob <- lmtest::coeftest(lm_final, vcov = vc)

list(
  selected_controls = selected_controls,
```

```r
    model = lm_final,
    robust_coefs = coefs_rob,
    lambda_y = lambda_y,
    d_lambdas = d_lambdas,
    y_constant = y_constant,
    d_constant = d_constant
  )
}

# Linear tests (same regressions as table 1)
calculate_tests <- function(model, model_type) {
  if (model_type == "any_treatment") {
    pval1 <- tryCatch(car::linearHypothesis(model, "RES05_gender = 0")$`Pr(>F)`[2], error = function(e)
    pval2 <- tryCatch(car::linearHypothesis(model, "INT_treatment:RES05_gender = 0")$`Pr(>F)`[2], error
    pval3 <- tryCatch(car::linearHypothesis(model, "INT_treatment = INT_treatment:RES05_gender")$`Pr(>F
    return(list(pval1 = round(pval1, 2), pval2 = round(pval2, 2), pval3 = round(pval3, 2)))
  } else {
    pval1 <- tryCatch(car::linearHypothesis(model, "INT_treatment_gender:RES05_gender = 0")$`Pr(>F)`[2]
    pval2 <- tryCatch(car::linearHypothesis(model, "INT_treatment_general:RES05_gender = 0")$`Pr(>F)`[2
    pval3 <- tryCatch(car::linearHypothesis(model, "INT_treatment_gender = INT_treatment_general")$`Pr(
    pval4 <- tryCatch(car::linearHypothesis(model, "INT_treatment_gender:RES05_gender = INT_treatment_g
    return(list(pval1 = round(pval1, 2), pval2 = round(pval2, 2), pval3 = round(pval3, 2), pval4 = roun
  }
}


# Loop on dependent variables

models_list <- list()
test_results <- list()
control_means <- list()

# Any treatment models
for (i in seq_along(incum_dep_vars1)) {
  dep_var <- incum_dep_vars1[i]


  # control mean
  control_means[[i]] <- data_filtered %>%
    filter(INT_treatment == 0, RES05_gender == 0) %>%
    summarise(mean = mean(!!sym(dep_var), na.rm = TRUE)) %>%
    pull(mean) %>% round(2)


  # treatment variables
  d_names_any <- c("INT_treatment", "RES05_gender", "INT_treatment:RES05_gender")

  # fe forced

  results <- double_lasso_fit(
    data_df = data_filtered,
    y_name = dep_var,
    d_names = d_names_any,
    x_candidates = full_controls_candidates,
    baseline_names = gpcontrols,
    fe_name = "district",
    se_cluster = data_filtered$district
  )


  models_list[[i]] <- results$model
  test_results[[i]] <- calculate_tests(results$model, "any_treatment")
}


# Gender / general models
for (i in seq_along(incum_dep_vars1)) {
  dep_var <- incum_dep_vars1[i]
  j <- i + length(incum_dep_vars1)

  control_means[[j]] <- control_means[[i]]

  d_names_gg <- c("INT_treatment_gender", "INT_treatment_general", "RES05_gender",
                  "INT_treatment_gender:RES05_gender", "INT_treatment_general:RES05_gender")

  results <- double_lasso_fit(
    data_df = data_filtered,
    y_name = dep_var,
    d_names = d_names_gg,
    x_candidates = full_controls_candidates,
    baseline_names = gpcontrols,
    fe_name = "district",
    se_cluster = data_filtered$district
  )

  models_list[[j]] <- results$model
  test_results[[j]] <- calculate_tests(results$model, "gender_general")
}


## OUTPUT ##

# Summary of treatment effects
treat_vars_any <- c("INT_treatment", "RES05_gender", "INT_treatment:RES05_gender")
treat_vars_gg  <- c("INT_treatment_gender", "INT_treatment_general",
                    "RES05_gender", "INT_treatment_gender:RES05_gender",
                    "INT_treatment_general:RES05_gender")
all_treat_vars <- union(treat_vars_any, treat_vars_gg)

tidy_list <- map(models_list, ~ broom::tidy(.x))
tidy_list <- Map(function(df, id) { df$model_id <- id; df }, tidy_list, seq_along(tidy_list))

treat_table <- bind_rows(tidy_list) %>%
  filter(term %in% all_treat_vars) %>%
  mutate(
    dep_var = rep(incum_dep_vars1, 2)[model_id],
    spec    = ifelse(model_id <= length(incum_dep_vars1), "any_treatment", "gender_general"),
    est_se  = sprintf("%.3f (%.3f)", estimate, std.error)
  ) %>%
  select(dep_var, spec, term, est_se, p.value)

cat("==== TREATMENT: COEFS AND SE ====\n")
```

```
## ==== TREATMENT: COEFS AND SE ====
```

```
print(treat_table, row.names = FALSE, n=Inf)
```

```
## # A tibble: 48 x 5
##    dep_var               spec           term                      est_se p.value
##    <chr>                 <chr>          <chr>                     <chr>    <dbl>
##  1 INC05_running         any_treatment  INT_treatment             -0.26~ 7.31e-3
##  2 INC05_running         any_treatment  RES05_gender              -0.32~ 2.46e-3
##  3 INC05_running         any_treatment  INT_treatment:RES05_gend~ 0.281~ 4.79e-2
##  4 INC05_voteshare       any_treatment  INT_treatment             -4.27~ 1.00e-1
##  5 INC05_voteshare       any_treatment  RES05_gender              -7.80~ 5.69e-3
##  6 INC05_voteshare       any_treatment  INT_treatment:RES05_gend~ 5.669~ 1.33e-1
##  7 INCSPOUSE05_running   any_treatment  INT_treatment             0.050~ 5.23e-1
##  8 INCSPOUSE05_running   any_treatment  RES05_gender              0.325~ 1.76e-4
##  9 INCSPOUSE05_running   any_treatment  INT_treatment:RES05_gend~ -0.24~ 2.89e-2
## 10 INCSPOUSE05_voteshare any_treatment  INT_treatment             0.167~ 9.32e-1
## 11 INCSPOUSE05_voteshare any_treatment  RES05_gender              5.709~ 7.33e-3
## 12 INCSPOUSE05_voteshare any_treatment  INT_treatment:RES05_gend~ -3.89~ 1.71e-1
## 13 INCOTHER05_running    any_treatment  INT_treatment             0.128~ 6.29e-2
## 14 INCOTHER05_running    any_treatment  RES05_gender              -0.14~ 5.09e-2
## 15 INCOTHER05_running    any_treatment  INT_treatment:RES05_gend~ 0.025~ 7.97e-1
## 16 INCOTHER05_voteshare  any_treatment  INT_treatment             1.669~ 2.77e-1
## 17 INCOTHER05_voteshare  any_treatment  RES05_gender              -1.79~ 2.76e-1
## 18 INCOTHER05_voteshare  any_treatment  INT_treatment:RES05_gend~ 2.316~ 2.99e-1
## 19 INC05_running         gender_general INT_treatment_gender      -0.19~ 1.35e-1
## 20 INC05_running         gender_general INT_treatment_general     -0.29~ 1.34e-2
## 21 INC05_running         gender_general RES05_gender              -0.31~ 3.42e-3
## 22 INC05_running         gender_general INT_treatment_gender:RES~ 0.245~ 1.58e-1
## 23 INC05_running         gender_general INT_treatment_general:RE~ 0.259~ 1.51e-1
## 24 INC05_voteshare       gender_general INT_treatment_gender      -0.63~ 8.47e-1
## 25 INC05_voteshare       gender_general INT_treatment_general     -7.07~ 2.45e-2
## 26 INC05_voteshare       gender_general RES05_gender              -8.22~ 3.62e-3
## 27 INC05_voteshare       gender_general INT_treatment_gender:RES~ 2.975~ 5.03e-1
## 28 INC05_voteshare       gender_general INT_treatment_general:RE~ 6.417~ 1.82e-1
## 29 INCSPOUSE05_running   gender_general INT_treatment_gender      -0.02~ 7.73e-1
## 30 INCSPOUSE05_running   gender_general INT_treatment_general     0.080~ 3.94e-1
## 31 INCSPOUSE05_running   gender_general RES05_gender              0.338~ 1.16e-4
## 32 INCSPOUSE05_running   gender_general INT_treatment_gender:RES~ -0.12~ 3.67e-1
## 33 INCSPOUSE05_running   gender_general INT_treatment_general:RE~ -0.36~ 1.24e-2
## 34 INCSPOUSE05_voteshare gender_general INT_treatment_gender      -0.47~ 8.50e-1
## 35 INCSPOUSE05_voteshare gender_general INT_treatment_general     1.180~ 6.16e-1
## 36 INCSPOUSE05_voteshare gender_general RES05_gender              5.657~ 7.90e-3
## 37 INCSPOUSE05_voteshare gender_general INT_treatment_gender:RES~ -1.35~ 6.86e-1
## 38 INCSPOUSE05_voteshare gender_general INT_treatment_general:RE~ -8.16~ 2.56e-2
## 39 INCOTHER05_running    gender_general INT_treatment_gender      0.111~ 2.05e-1
## 40 INCOTHER05_running    gender_general INT_treatment_general     0.146~ 7.62e-2
## 41 INCOTHER05_running    gender_general RES05_gender             -0.16~ 3.17e-2
## 42 INCOTHER05_running    gender_general INT_treatment_gender:RES~ 0.066~ 5.77e-1
## 43 INCOTHER05_running    gender_general INT_treatment_general:RE~ -0.01~ 9.20e-1
## 44 INCOTHER05_voteshare  gender_general INT_treatment_gender      0.927~ 6.40e-1
## 45 INCOTHER05_voteshare  gender_general INT_treatment_general     2.508~ 1.84e-1
## 46 INCOTHER05_voteshare  gender_general RES05_gender             -2.06~ 2.21e-1
## 47 INCOTHER05_voteshare  gender_general INT_treatment_gender:RES~ 5.018~ 6.35e-2
## 48 INCOTHER05_voteshare  gender_general INT_treatment_general:RE~ -1.22~ 6.73e-1
```

```r
cat("\n\n")
```

```r
print(treat_table, n = Inf, width = Inf)
```

```
## # A tibble: 48 x 5
##    dep_var                spec           term
##    <chr>                  <chr>          <chr>
##  1 INC05_running          any_treatment  INT_treatment
##  2 INC05_running          any_treatment  RES05_gender
##  3 INC05_running          any_treatment  INT_treatment:RES05_gender
##  4 INC05_voteshare        any_treatment  INT_treatment
##  5 INC05_voteshare        any_treatment  RES05_gender
##  6 INC05_voteshare        any_treatment  INT_treatment:RES05_gender
##  7 INCSPOUSE05_running    any_treatment  INT_treatment
##  8 INCSPOUSE05_running    any_treatment  RES05_gender
##  9 INCSPOUSE05_running    any_treatment  INT_treatment:RES05_gender
## 10 INCSPOUSE05_voteshare any_treatment  INT_treatment
## 11 INCSPOUSE05_voteshare any_treatment  RES05_gender
## 12 INCSPOUSE05_voteshare any_treatment  INT_treatment:RES05_gender
## 13 INCOTHER05_running    any_treatment  INT_treatment
## 14 INCOTHER05_running    any_treatment  RES05_gender
## 15 INCOTHER05_running    any_treatment  INT_treatment:RES05_gender
## 16 INCOTHER05_voteshare  any_treatment  INT_treatment
## 17 INCOTHER05_voteshare  any_treatment  RES05_gender
## 18 INCOTHER05_voteshare  any_treatment  INT_treatment:RES05_gender
## 19 INC05_running          gender_general INT_treatment_gender
## 20 INC05_running          gender_general INT_treatment_general
## 21 INC05_running          gender_general RES05_gender
## 22 INC05_running          gender_general INT_treatment_gender:RES05_gender
## 23 INC05_running          gender_general INT_treatment_general:RES05_gender
## 24 INC05_voteshare        gender_general INT_treatment_gender
## 25 INC05_voteshare        gender_general INT_treatment_general
## 26 INC05_voteshare        gender_general RES05_gender
## 27 INC05_voteshare        gender_general INT_treatment_gender:RES05_gender
## 28 INC05_voteshare        gender_general INT_treatment_general:RES05_gender
## 29 INCSPOUSE05_running    gender_general INT_treatment_gender
## 30 INCSPOUSE05_running    gender_general INT_treatment_general
## 31 INCSPOUSE05_running    gender_general RES05_gender
## 32 INCSPOUSE05_running    gender_general INT_treatment_gender:RES05_gender
## 33 INCSPOUSE05_running    gender_general INT_treatment_general:RES05_gender
## 34 INCSPOUSE05_voteshare gender_general INT_treatment_gender
## 35 INCSPOUSE05_voteshare gender_general INT_treatment_general
## 36 INCSPOUSE05_voteshare gender_general RES05_gender
## 37 INCSPOUSE05_voteshare gender_general INT_treatment_gender:RES05_gender
## 38 INCSPOUSE05_voteshare gender_general INT_treatment_general:RES05_gender
## 39 INCOTHER05_running    gender_general INT_treatment_gender
## 40 INCOTHER05_running    gender_general INT_treatment_general
## 41 INCOTHER05_running    gender_general RES05_gender
## 42 INCOTHER05_running    gender_general INT_treatment_gender:RES05_gender
## 43 INCOTHER05_running    gender_general INT_treatment_general:RES05_gender
## 44 INCOTHER05_voteshare  gender_general INT_treatment_gender
## 45 INCOTHER05_voteshare  gender_general INT_treatment_general
## 46 INCOTHER05_voteshare  gender_general RES05_gender
## 47 INCOTHER05_voteshare  gender_general INT_treatment_gender:RES05_gender
## 48 INCOTHER05_voteshare  gender_general INT_treatment_general:RES05_gender
##    est_se          p.value
##    <chr>             <dbl>
##  1 -0.266 (0.097) 0.00731
##  2 -0.322 (0.104) 0.00246
##  3 0.281 (0.141)  0.0479
##  4 -4.277 (2.583) 0.100
##  5 -7.806 (2.776) 0.00569
##  6 5.669 (3.749)  0.133
##  7 0.050 (0.078)  0.523
##  8 0.325 (0.084)  0.000176
##  9 -0.249 (0.113) 0.0289
## 10 0.167 (1.950)  0.932
## 11 5.709 (2.095)  0.00733
## 12 -3.895 (2.830) 0.171
## 13 0.128 (0.068)  0.0629
## 14 -0.145 (0.074) 0.0509
## 15 0.025 (0.099)  0.797
## 16 1.669 (1.529)  0.277
## 17 -1.796 (1.643) 0.276
## 18 2.316 (2.220)  0.299
## 19 -0.191 (0.127) 0.135
## 20 -0.295 (0.118) 0.0134
## 21 -0.319 (0.107) 0.00342
## 22 0.245 (0.172)  0.158
## 23 0.259 (0.180)  0.151
## 24 -0.633 (3.273) 0.847
## 25 -7.072 (3.106) 0.0245
## 26 -8.223 (2.773) 0.00362
## 27 2.975 (4.431)  0.503
## 28 6.417 (4.781)  0.182
## 29 -0.029 (0.100) 0.773
## 30 0.080 (0.094)  0.394
## 31 0.338 (0.085)  0.000116
## 32 -0.123 (0.136) 0.367
## 33 -0.363 (0.143) 0.0124
## 34 -0.470 (2.473) 0.850
## 35 1.180 (2.347)  0.616
## 36 5.657 (2.096)  0.00790
## 37 -1.359 (3.348) 0.686
## 38 -8.161 (3.613) 0.0256
## 39 0.111 (0.087)  0.205
## 40 0.146 (0.082)  0.0762
## 41 -0.161 (0.074) 0.0317
## 42 0.066 (0.118)  0.577
## 43 -0.013 (0.125) 0.920
## 44 0.927 (1.980)  0.640
## 45 2.508 (1.879)  0.184
## 46 -2.063 (1.678) 0.221
## 47 5.018 (2.680)  0.0635
## 48 -1.222 (2.892) 0.673
```

```r
# comparison of the two selected lists
baseline_controls <- gpcontrols

selected_controls_list <- map(models_list, ~ {
  vars <- names(coef(.x))[-1]
  setdiff(vars, all_treat_vars)
})

controls_diff <- tibble::tibble(
  dep_var     = rep(incum_dep_vars1, 2),
  spec        = c(rep("any_treatment", length(incum_dep_vars1)),
                  rep("gender_general", length(incum_dep_vars1))),
  selected    = selected_controls_list
) %>%
  mutate(
    kept    = map(selected, ~ intersect(.x, baseline_controls)),
    dropped = map(selected, ~ setdiff(baseline_controls, .x)),
    added   = map(selected, ~ setdiff(.x, baseline_controls)),
    n_selected = map_int(selected, length)
  )

cat("==== COMPARISON OF CONTROLS (baseline vs double lasso) ====\n")
```

```
## ==== COMPARISON OF CONTROLS (baseline vs double lasso) ====
```

```r
for (r in 1:nrow(controls_diff)) {
  cat("\n--- DV:", controls_diff$dep_var[r],
      " | Spec:", controls_diff$spec[r], " ---\n")
  cat("# Selected controls =", controls_diff$n_selected[r], "\n")
  cat("Kept (baseline)   :", paste(controls_diff$kept[[r]],    collapse = ", "), "\n")
  cat("Dropped (baseline):", paste(controls_diff$dropped[[r]], collapse = ", "), "\n")
  cat("Added (non-base)  :", paste(controls_diff$added[[r]],   collapse = ", "), "\n")
}
```

```
##
## --- DV: INC05_running  | Spec: any_treatment  ---
## # Selected controls = 18
## Kept (baseline)   : GP_population, RES00_obc, RES10_obc, RES10_st, RES05_sc, GP_nbvillages, RES00_ge
## Dropped (baseline): GP_lit, GP_sc, GP_st, RES10_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INC05_voteshare  | Spec: any_treatment  ---
## # Selected controls = 16
## Kept (baseline)   : GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES05_obc, RES05_st, RES00_obc,
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, RES10_sc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, CENSUS_VD2001_edu_fac, fe_BHARA
##
## --- DV: INCSPOUSE05_running  | Spec: any_treatment  ---
## # Selected controls = 16
## Kept (baseline)   : GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES05_obc, RES05_st, RES00_obc,
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, RES10_sc, RES05_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCSPOUSE05_voteshare  | Spec: any_treatment  ---
## # Selected controls = 16
## Kept (baseline)   : GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES05_obc, RES05_st, RES00_obc,
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, RES10_sc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, CENSUS_VD2001_edu_fac, fe_BHARA
##
## --- DV: INCOTHER05_running  | Spec: any_treatment  ---
## # Selected controls = 16
## Kept (baseline)   : GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES05_obc, RES05_st, RES00_obc,
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, RES10_sc, RES05_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_voteshare  | Spec: any_treatment  ---
## # Selected controls = 16
## Kept (baseline)   : GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES05_obc, RES05_st, RES00_obc,
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, RES10_sc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, CENSUS_VD2001_edu_fac, fe_BHARA
##
## --- DV: INC05_running  | Spec: gender_general  ---
## # Selected controls = 19
## Kept (baseline)   : GP_population, RES00_obc, RES10_obc, RES10_st, RES05_sc, GP_sc, RES00_gender, RES
## Dropped (baseline): GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_pow
##
## --- DV: INC05_voteshare  | Spec: gender_general  ---
## # Selected controls = 18
## Kept (baseline)   : GP_sc, RES00_gender, RES00_st, RES10_st, RES05_sc, RES05_st, RES00_obc, RES05_ob
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_power_dom, CENSUS_VD2001_medi_fac, fe_BHA
##
## --- DV: INCSPOUSE05_running  | Spec: gender_general  ---
## # Selected controls = 18
## Kept (baseline)   : GP_sc, RES00_gender, RES00_st, RES10_st, RES05_sc, RES05_st, RES00_obc, RES05_ob
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_pow
##
## --- DV: INCSPOUSE05_voteshare  | Spec: gender_general  ---
## # Selected controls = 18
## Kept (baseline)   : GP_sc, RES00_gender, RES00_st, RES10_st, RES05_sc, RES05_st, RES00_obc, RES05_ob
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_power_dom, CENSUS_VD2001_medi_fac, fe_BHA
##
## --- DV: INCOTHER05_running  | Spec: gender_general  ---
## # Selected controls = 18
## Kept (baseline)   : GP_sc, RES00_gender, RES00_st, RES10_st, RES05_sc, RES05_st, RES00_obc, RES05_ob
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_pow
##
## --- DV: INCOTHER05_voteshare  | Spec: gender_general  ---
## # Selected controls = 18
## Kept (baseline)   : GP_sc, RES00_gender, RES00_st, RES10_st, RES05_sc, RES05_st, RES00_obc, RES05_ob
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_sc, RES10_sc
## Added (non-base)  : CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_power_dom, CENSUS_VD2001_medi_fac, fe_BHA
```

```
cat("\n==== END ====\n")
```

```
##
## ==== END ====
```

Replication of Table 2 using double lasso.

```
###################################### TABLE 2 ####################################

# Data and macros
## Baseline list (initially chosen controls)
gpcontrols <- c("GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
                "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
                "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st")

## Full list of candidates
full_controls_candidates <- c(
  # Baseline
  "GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
  "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
  "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st",
  # Other variables - from the census
  "CENSUS_PCA2001_tot_pop", "CENSUS_PCA2001_tot_lit", "CENSUS_PCA2001_tot_sc",
  "CENSUS_PCA2001_tot_st", "CENSUS_PCA2001_tot_aglb", "CENSUS_PCA2001_tot_nm_hh",
  "CENSUS_PCA2001_tot_cult", "CENSUS_VD2001_power_dom", "CENSUS_VD2001_drnk_wat_f",
  "CENSUS_VD2001_edu_fac", "CENSUS_VD2001_medi_fac"
)

# Dependent variables
incum_dep_vars2 <- c(
  "INC05_running", "INC05_voteshare", "INC05_won",
  "INCSPOUSE05_running", "INCSPOUSE05_voteshare", "INCSPOUSE05_won",
  "INCOTHER05_running", "INCOTHER05_voteshare", "INCOTHER05_won"
)

# Performance indices
indices <- c("index_empl_svy_0", "index_empl_svy_1", "index_empl_svy_2", "index_empl_svy_3")



# Loading and filtering the data
data <- haven::read_dta("~/work/Electoral data cleaned.dta")
data_filtered <- data %>%
  filter(RES10_gender == 0, SAMPLE_hhsurvey == 1, GP_tag == 1, INC05_can_run == 1) %>%
  mutate(
    FAMnotINC05_running   = INCorFAM05_running  - INC05_running,
    FAMnotINC05_voteshare = INCorFAM05_voteshare - INC05_voteshare,
    FAMnotINC05_won       = INCorFAM05_won - INC05_won,
    # Generating the performance indices
    index_empl_svy_0 = rowMeans(select(., std_HH_NREGA, std_HH_NREGA_unmet_demand_m, std_HH_NREGA_unmet
    index_empl_svy_1 = rowMeans(select(., std_HH_NREGA_unmet_demand, std_HH_NREGA_unmet_demand_m, std_H
    index_empl_svy_2 = rowMeans(select(., std_HH_NREGA, std_HH_NREGA_work_m, std_HH_NREGA_work_f), na.r
    index_empl_svy_3 = rowMeans(select(., std_HH_NREGA_unmet_demand_m, std_HH_NREGA_unmet_demand_f), na
  )

# Checking candidate controls
data_filtered %>% select(all_of(full_controls_candidates)) %>% summary()
```

```
##  GP_population       GP_lit           GP_sc              GP_st
##  Min.   : 3250   Min.   :0.3205   Min.   :0.009154   Min.   :0.000000
##  1st Qu.: 4610   1st Qu.:0.5096   1st Qu.:0.150252   1st Qu.:0.000000
##  Median : 5550   Median :0.5599   Median :0.232531   Median :0.001974
##  Mean   : 5762   Mean   :0.5478   Mean   :0.226674   Mean   :0.089329
##  3rd Qu.: 6658   3rd Qu.:0.6011   3rd Qu.:0.301907   3rd Qu.:0.124034
##  Max.   :12274   Max.   :0.6856   Max.   :0.548745   Max.   :0.705038
##  GP_nbvillages     RES00_gender      RES00_obc         RES00_sc
##  Min.   : 1.000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.: 3.000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median : 4.000   Median :0.0000   Median :0.0000   Median :0.0000
##  Mean   : 4.493   Mean   :0.2895   Mean   :0.1711   Mean   :0.2566
##  3rd Qu.: 5.000   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000
##  Max.   :12.000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##     RES00_st        RES10_obc         RES10_sc          RES10_st
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.0000   Median :0.0000   Median :0.00000   Median :0.00000
##  Mean   :0.1447   Mean   :0.1316   Mean   :0.01316   Mean   :0.05263
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
##    RES05_obc         RES05_sc         RES05_st       CENSUS_PCA2001_tot_pop
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   : 2782
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.: 3950
##  Median :0.0000   Median :0.0000   Median :0.00000   Median : 4711
##  Mean   :0.1184   Mean   :0.2368   Mean   :0.07237   Mean   : 5303
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.: 5735
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :25548
##  CENSUS_PCA2001_tot_lit CENSUS_PCA2001_tot_sc CENSUS_PCA2001_tot_st
##  Min.   :  920          Min.   :   71         Min.   :   0.0
##  1st Qu.: 1801          1st Qu.:  629         1st Qu.:   0.0
##  Median : 2252          Median : 1033         Median :   5.5
##  Mean   : 2500          Mean   : 1225         Mean   : 485.8
##  3rd Qu.: 2877          3rd Qu.: 1579         3rd Qu.: 619.8
##  Max.   :15330          Max.   :11400         Max.   :6915.0
##  CENSUS_PCA2001_tot_aglb CENSUS_PCA2001_tot_nm_hh CENSUS_PCA2001_tot_cult
##  Min.   :  3.0           Min.   : 370.0           Min.   : 438.0
##  1st Qu.: 37.0           1st Qu.: 571.5           1st Qu.: 779.8
##  Median : 67.5           Median : 693.0           Median : 986.5
##  Mean   : 87.7           Mean   : 776.6           Mean   :1062.6
##  3rd Qu.:116.2           3rd Qu.: 832.8           3rd Qu.:1198.5
##  Max.   :351.0           Max.   :3786.0           Max.   :4182.0
##  CENSUS_VD2001_power_dom CENSUS_VD2001_drnk_wat_f CENSUS_VD2001_edu_fac
##  Min.   :0.0000          Min.   :0.5000           Min.   :0.4000
##  1st Qu.:0.0000          1st Qu.:1.0000           1st Qu.:0.8750
##  Median :0.2111          Median :1.0000           Median :1.0000
##  Mean   :0.3223          Mean   :0.9604           Mean   :0.9275
##  3rd Qu.:0.6000          3rd Qu.:1.0000          3rd Qu.:1.0000
##  Max.   :1.0000          Max.   :1.0000          Max.   :1.0000
##  CENSUS_VD2001_medi_fac
##  Min.   :0.0000
##  1st Qu.:0.1917
##  Median :0.4000
##  Mean   :0.4267
##  3rd Qu.:0.6667
##  Max.   :1.0000
```

```r
## FUNCTIONS ##

# Function to extract the names of non zero coefficients
get_nonzero_names <- function(cf, drop = "(Intercept)") {
  cf_mat <- as.matrix(cf)
  nz <- which(cf_mat != 0)
  vars <- rownames(cf_mat)[nz]
  setdiff(vars, drop)
}

# Double lasso function
double_lasso_fit <- function(data_df, y_name, d_names, x_candidates, baseline_names = NULL, fe_name = "
  # fixed effects
  if (!is.null(fe_name)) {
    fe_vec <- data_df[[fe_name]]
    fe_mat <- model.matrix(~ factor(fe_vec) - 1)
    colnames(fe_mat) <- paste0("fe_", levels(factor(fe_vec)))
  } else {
    fe_mat <- NULL
  }

  # X : candidates
  X_all <- data_df %>% dplyr::select(all_of(x_candidates))
  X_mat <- model.matrix(~ . - 1, data = X_all)

  # Filter NA
  keep_rows_X <- complete.cases(X_mat)
  if (!is.null(fe_mat)) {
    keep_rows_FE <- complete.cases(fe_mat)
    keep_rows <- keep_rows_X & keep_rows_FE
    X_mat <- X_mat[keep_rows, , drop = FALSE]
    fe_mat <- fe_mat[keep_rows, , drop = FALSE]
  } else {
    X_mat <- X_mat[keep_rows_X, , drop = FALSE]
    keep_rows <- keep_rows_X
  }

  baseline_idx <- if (!is.null(baseline_names)) {
    which(colnames(X_mat) %in% baseline_names)
  } else integer(0)

  if (!is.null(fe_mat)) {
    X_mat <- cbind(X_mat, fe_mat)
    fe_idx <- (ncol(X_mat) - ncol(fe_mat) + 1):ncol(X_mat)
```

```r
} else {
  fe_idx <- integer(0)
}

# Y and D
missing_d <- setdiff(d_names, names(data_df))
if (length(missing_d) > 0) {
  for (nm in missing_d) {
    parts <- strsplit(nm, ":", fixed = TRUE)[[1]]
    stopifnot(all(parts %in% names(data_df)))
    data_df[[nm]] <- data_df[[parts[1]]] * data_df[[parts[2]]]
  }
}

y_vec <- as.numeric(data_df[[y_name]])
D_mat <- as.matrix(data_df %>% dplyr::select(all_of(d_names)))
if (is.null(ncol(D_mat))) D_mat <- matrix(D_mat, ncol = 1)

# Filter y_vec and D_mat
y_vec <- y_vec[keep_rows]
D_mat <- D_mat[keep_rows, , drop = FALSE]
if (!is.null(se_cluster)) se_cluster <- se_cluster[keep_rows]

# checks
y_constant <- length(unique(y_vec)) < 2
d_constant <- apply(D_mat, 2, function(z) length(unique(z)) < 2)

# Y ~ D + X
sel_y <- character(0); lambda_y <- NA
if (!y_constant) {
  X_y <- cbind(D_mat, X_mat)
  pf_y <- c(rep(0, ncol(D_mat)), rep(1, ncol(X_mat)))
  if (length(fe_idx) > 0) pf_y[ncol(D_mat) + fe_idx] <- 0

  if (all(pf_y == 0)) {
    sel_y <- character(0)
    lambda_y <- NA
  } else {
    set.seed(123)
    fit1 <- tryCatch({
      cv.glmnet(X_y, y_vec, alpha = 1, family = family, penalty.factor = pf_y)
    }, error = function(e) NULL)

    if (!is.null(fit1)) {
      coef1 <- coef(fit1, s = "lambda.min")
      cf_mat1 <- as.matrix(coef1)
      nz1 <- which(cf_mat1 != 0)
      sel_y <- rownames(cf_mat1)[nz1]
      sel_y <- setdiff(sel_y, c("(Intercept)", colnames(D_mat)))
      lambda_y <- fit1$lambda.min
    } else {
      sel_y <- character(0)
      lambda_y <- NA
    }
  }
}

# D_k ~ X
pf_d <- rep(1, ncol(X_mat))
if (length(baseline_idx) > 0) pf_d[baseline_idx] <- 1
if (length(fe_idx) > 0) pf_d[fe_idx] <- 0

sel_d <- character(0)
d_lambdas <- rep(NA, ncol(D_mat))

if (!all(pf_d == 0)) {
  for (k in seq_len(ncol(D_mat))) {
    if (d_constant[k]) next
    set.seed(123)
    fitk <- tryCatch({
      cv.glmnet(X_mat, D_mat[, k], alpha = 1, family = "gaussian", penalty.factor = pf_d)
    }, error = function(e) NULL)

    if (!is.null(fitk)) {
      coefk <- coef(fitk, s = "lambda.min")
      cf_matk <- as.matrix(coefk)
      nz_k <- which(cf_matk != 0)
      sel_k <- rownames(cf_matk)[nz_k]
      sel_k <- setdiff(sel_k, "(Intercept)")
      sel_d <- union(sel_d, sel_k)
      d_lambdas[k] <- fitk$lambda.min
    }
  }
}

# Union and final regression
selected_controls <- union(sel_y, sel_d)
if (length(fe_idx) > 0) selected_controls <- union(selected_controls, colnames(X_mat)[fe_idx])

X_final <- X_mat[, selected_controls, drop = FALSE]
final_df <- data.frame(Y = y_vec, D_mat, X_final)
colnames(final_df)[2:(1 + ncol(D_mat))] <- colnames(D_mat)

rhs <- paste(c(colnames(D_mat), colnames(X_final)), collapse = " + ")
f_final <- as.formula(paste("Y ~", rhs))
lm_final <- lm(f_final, data = final_df)

vc <- if (!is.null(se_cluster)) {
  sandwich::vcovCL(lm_final, cluster = se_cluster)
} else {
  sandwich::vcovHC(lm_final, type = "HC1")
}

coefs_rob <- lmtest::coeftest(lm_final, vcov = vc)

list(
```

```r
    selected_controls = selected_controls,
    model = lm_final,
    robust_coefs = coefs_rob,
    lambda_y = lambda_y,
    d_lambdas = d_lambdas,
    y_constant = y_constant,
    d_constant = d_constant
  )
}

# Function to compute marginal effects and do the tests
calculate_effects <- function(model, index_name, index_mean, index_sd) {
  # marginal effects
  effect_average <- coef(model)["INT_treatment"] + coef(model)[paste0("INT_treatment:", index_name)] * i
  effect_good <- coef(model)["INT_treatment"] + coef(model)[paste0("INT_treatment:", index_name)] * (ind
  effect_bad <- coef(model)["INT_treatment"] + coef(model)[paste0("INT_treatment:", index_name)] * (inde

  # Wald tests
  test_1 <- tryCatch({
    waldtest(model, c(paste0("INT_treatment + INT_treatment:", index_name, " = 0"),
                      paste0(index_name, "=", index_mean)))
  }, error = function(e) NULL)

  test_2 <- tryCatch({
    waldtest(model, c(paste0("INT_treatment + INT_treatment:", index_name, " = 0")))
  }, error = function(e) NULL)

  pval_1 <- if (!is.null(test_1)) round(test_1$p.value, 2) else NA
  pval_2 <- if (!is.null(test_2)) round(test_2$p.value, 2) else NA

  list(
    effect_average = effect_average,
    effect_good = effect_good,
    effect_bad = effect_bad,
    pval_1 = pval_1,
    pval_2 = pval_2
  )
}


## LOOP FOR THE REGRESSIONS ##

models_list <- list()
control_means <- list()
effects_list <- list()
pvals_1 <- list()
pvals_2 <- list()


i <- 0
for (x in 0:1) {  # loop on 2005 gender reservation
  for (dep_var in incum_dep_vars2) {
    for (index in indices) {
      i <- i + 1

      # control mean
      control_mean <- data_filtered %>%
        filter(INT_treatment == 0 & RES05_gender == x) %>%
        summarise(mean = mean(!!sym(dep_var), na.rm = TRUE)) %>%
        pull(mean) %>%
        round(2)
      control_means[[i]] <- control_mean

      # index stats
      index_stats <- data_filtered %>%
        filter(RES05_gender == x) %>%
        summarise(mean = mean(!!sym(index), na.rm = TRUE), sd = sd(!!sym(index), na.rm = TRUE))
      index_mean <- round(index_stats$mean, 2)
      index_sd <- round(index_stats$sd, 2)

      # interaction variables
      data_filtered <- data_filtered %>%
        mutate(
          TEMP_index = get(index),
          TEMP_X_anytr_index = INT_treatment * get(index)
        )

      # checking varaibles
      all_vars <- c(dep_var, "INT_treatment", "TEMP_index", "TEMP_X_anytr_index", gpcontrols, "district
      if (all(all_vars %in% names(data_filtered))) {
        # checking clusters
        filtered_data <- data_filtered %>% filter(RES05_gender == x)
        if (length(unique(filtered_data$district)) < 2) {
          message("Not enough clusters for RES05_gender = ", x, " and dep_var = ", dep_var)
          next
        }

        # treatment variables
        d_names <- c("INT_treatment", "TEMP_index", "TEMP_X_anytr_index")

        # Double lasso
        results <- double_lasso_fit(
          data_df = filtered_data,
          y_name = dep_var,
          d_names = d_names,
          x_candidates = full_controls_candidates,
          baseline_names = gpcontrols,
          fe_name = "district",
          se_cluster = filtered_data$district
        )

        models_list[[i]] <- results$model

        # compute effects and tests
        effects <- calculate_effects(results$model, "TEMP_index", index_mean, index_sd)
        effects_list[[i]] <- effects
        pvals_1[[i]] <- effects$pval_1
        pvals_2[[i]] <- effects$pval_2

        # display the effects
        cat("Effects on outcome", dep_var, "with index", index, "\n")
        cat("Effect for average performance:", effects$effect_average, "\n")
        cat("Effect for +1 sd performance:", effects$effect_good, "\n")
        cat("Effect for -1 sd performance:", effects$effect_bad, "\n")
      } else {
        message("Missing variables for ", dep_var, "with index", index)
      }
    }
  }
}
```

```
## Effects on outcome INC05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INC05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCSPOUSE05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_running with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_voteshare with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_0
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_1
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_2
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
## Effects on outcome INCOTHER05_won with index index_empl_svy_3
## Effect for average performance: NA
## Effect for +1 sd performance: NA
## Effect for -1 sd performance: NA
```

```
## OUTPUT ##

# Table of treatment variables' coefficients
treat_vars <- c("INT_treatment", "TEMP_index", "TEMP_X_anytr_index")
tidy_list <- map(models_list, ~ broom::tidy(.x))
tidy_list <- Map(function(df, id) { df$model_id <- id; df }, tidy_list, seq_along(tidy_list))
treat_table <- bind_rows(tidy_list) %>%
  filter(term %in% treat_vars) %>%
  mutate(
    dep_var = rep(incum_dep_vars2, each = length(indices) * 2)[model_id],
    index = rep(rep(indices, each = 2), length(incum_dep_vars2))[model_id],
    RES05_gender = rep(rep(c(0, 1), each = length(indices)), length(incum_dep_vars2))[model_id],
    est_se = sprintf("%.3f (%.3f)", estimate, std.error)
  ) %>%
  select(dep_var, index, RES05_gender, term, est_se, p.value)

cat("==== TREATMENT COEFFICIENTS ====\n")
```

```
## ==== TREATMENT COEFFICIENTS ====
```

```r
print(treat_table, row.names = FALSE, n=Inf)
```

```
## # A tibble: 216 x 6
##     dep_var               index            RES05_gender term      est_se p.value
##     <chr>                 <chr>                   <dbl> <chr>     <chr>    <dbl>
##   1 INC05_running         index_empl_svy_0            0 INT_trea~ -0.26~ 0.0111
##   2 INC05_running         index_empl_svy_0            0 TEMP_ind~ -0.06~ 0.329
##   3 INC05_running         index_empl_svy_0            0 TEMP_X_a~ 0.229~ 0.0407
##   4 INC05_running         index_empl_svy_0            0 INT_trea~ -0.27~ 0.0113
##   5 INC05_running         index_empl_svy_0            0 TEMP_ind~ -0.03~ 0.816
##   6 INC05_running         index_empl_svy_0            0 TEMP_X_a~ 0.196~ 0.322
##   7 INC05_running         index_empl_svy_1            0 INT_trea~ -0.26~ 0.0132
##   8 INC05_running         index_empl_svy_1            0 TEMP_ind~ 0.003~ 0.970
##   9 INC05_running         index_empl_svy_1            0 TEMP_X_a~ 0.092~ 0.386
##  10 INC05_running         index_empl_svy_1            0 INT_trea~ -0.29~ 0.00729
##  11 INC05_running         index_empl_svy_1            0 TEMP_ind~ -0.07~ 0.284
##  12 INC05_running         index_empl_svy_1            0 TEMP_X_a~ 0.238~ 0.0329
##  13 INC05_running         index_empl_svy_2            1 INT_trea~ -5.10~ 0.0960
##  14 INC05_running         index_empl_svy_2            1 TEMP_ind~ -3.53~ 0.0638
##  15 INC05_running         index_empl_svy_2            1 TEMP_X_a~ 7.641~ 0.0200
##  16 INC05_running         index_empl_svy_2            1 INT_trea~ -2.78~ 0.390
##  17 INC05_running         index_empl_svy_2            1 TEMP_ind~ -1.52~ 0.709
##  18 INC05_running         index_empl_svy_2            1 TEMP_X_a~ 4.375~ 0.454
##  19 INC05_running         index_empl_svy_3            1 INT_trea~ -2.98~ 0.358
##  20 INC05_running         index_empl_svy_3            1 TEMP_ind~ -2.83~ 0.173
##  21 INC05_running         index_empl_svy_3            1 TEMP_X_a~ 2.233~ 0.482
##  22 INC05_running         index_empl_svy_3            1 INT_trea~ -6.27~ 0.0317
##  23 INC05_running         index_empl_svy_3            1 TEMP_ind~ -1.29~ 0.493
##  24 INC05_running         index_empl_svy_3            1 TEMP_X_a~ 5.722~ 0.0568
##  25 INC05_voteshare       index_empl_svy_0            0 INT_trea~ -0.01~ 0.745
##  26 INC05_voteshare       index_empl_svy_0            0 TEMP_ind~ -0.02~ 0.463
##  27 INC05_voteshare       index_empl_svy_0            0 TEMP_X_a~ -0.02~ 0.737
##  28 INC05_voteshare       index_empl_svy_0            0 INT_trea~ -0.00~ 0.960
##  29 INC05_voteshare       index_empl_svy_0            0 TEMP_ind~ -0.09~ 0.193
##  30 INC05_voteshare       index_empl_svy_0            0 TEMP_X_a~ 0.009~ 0.930
##  31 INC05_voteshare       index_empl_svy_1            0 INT_trea~ -0.01~ 0.799
##  32 INC05_voteshare       index_empl_svy_1            0 TEMP_ind~ -0.03~ 0.423
##  33 INC05_voteshare       index_empl_svy_1            0 TEMP_X_a~ -0.03~ 0.519
##  34 INC05_voteshare       index_empl_svy_1            0 INT_trea~ -0.02~ 0.669
##  35 INC05_voteshare       index_empl_svy_1            0 TEMP_ind~ -0.06~ 0.125
##  36 INC05_voteshare       index_empl_svy_1            0 TEMP_X_a~ 0.012~ 0.844
##  37 INC05_voteshare       index_empl_svy_2            1 INT_trea~ 0.002~ 0.971
##  38 INC05_voteshare       index_empl_svy_2            1 TEMP_ind~ -0.01~ 0.684
##  39 INC05_voteshare       index_empl_svy_2            1 TEMP_X_a~ 0.002~ 0.973
##  40 INC05_voteshare       index_empl_svy_2            1 INT_trea~ 0.010~ 0.864
##  41 INC05_voteshare       index_empl_svy_2            1 TEMP_ind~ -0.01~ 0.885
##  42 INC05_voteshare       index_empl_svy_2            1 TEMP_X_a~ -0.08~ 0.445
##  43 INC05_voteshare       index_empl_svy_3            1 INT_trea~ 0.012~ 0.834
##  44 INC05_voteshare       index_empl_svy_3            1 TEMP_ind~ -0.00~ 0.989
##  45 INC05_voteshare       index_empl_svy_3            1 TEMP_X_a~ -0.01~ 0.749
##  46 INC05_voteshare       index_empl_svy_3            1 INT_trea~ 0.011~ 0.863
##  47 INC05_voteshare       index_empl_svy_3            1 TEMP_ind~ -0.01~ 0.819
##  48 INC05_voteshare       index_empl_svy_3            1 TEMP_X_a~ 0.021~ 0.750
##  49 INC05_won             index_empl_svy_0            0 INT_trea~ -1.03~ 0.400
##  50 INC05_won             index_empl_svy_0            0 TEMP_ind~ -0.97~ 0.207
##  51 INC05_won             index_empl_svy_0            0 TEMP_X_a~ 0.694~ 0.596
##  52 INC05_won             index_empl_svy_0            0 INT_trea~ -0.70~ 0.568
##  53 INC05_won             index_empl_svy_0            0 TEMP_ind~ -0.74~ 0.632
##  54 INC05_won             index_empl_svy_0            0 TEMP_X_a~ -0.45~ 0.838
##  55 INC05_won             index_empl_svy_1            0 INT_trea~ -0.76~ 0.530
##  56 INC05_won             index_empl_svy_1            0 TEMP_ind~ -0.54~ 0.486
##  57 INC05_won             index_empl_svy_1            0 TEMP_X_a~ 0.282~ 0.813
##  58 INC05_won             index_empl_svy_1            0 INT_trea~ -0.73~ 0.567
##  59 INC05_won             index_empl_svy_1            0 TEMP_ind~ -0.81~ 0.341
##  60 INC05_won             index_empl_svy_1            0 TEMP_X_a~ 0.960~ 0.471
##  61 INC05_won             index_empl_svy_2            1 INT_trea~ -0.02~ 0.349
##  62 INC05_won             index_empl_svy_2            1 TEMP_ind~ -0.02~ 0.199
##  63 INC05_won             index_empl_svy_2            1 TEMP_X_a~ 0.021~ 0.472
##  64 INC05_won             index_empl_svy_2            1 INT_trea~ -0.01~ 0.484
##  65 INC05_won             index_empl_svy_2            1 TEMP_ind~ -0.03~ 0.367
##  66 INC05_won             index_empl_svy_2            1 TEMP_X_a~ 0.030~ 0.548
##  67 INC05_won             index_empl_svy_3            1 INT_trea~ -0.02~ 0.395
##  68 INC05_won             index_empl_svy_3            1 TEMP_ind~ -0.01~ 0.391
##  69 INC05_won             index_empl_svy_3            1 TEMP_X_a~ 0.014~ 0.608
##  70 INC05_won             index_empl_svy_3            1 INT_trea~ -0.01~ 0.503
##  71 INC05_won             index_empl_svy_3            1 TEMP_ind~ -0.02~ 0.145
##  72 INC05_won             index_empl_svy_3            1 TEMP_X_a~ 0.029~ 0.338
##  73 INCSPOUSE05_running   index_empl_svy_0            0 INT_trea~ 0.156~ 0.0461
##  74 INCSPOUSE05_running   index_empl_svy_0            0 TEMP_ind~ 0.032~ 0.504
##  75 INCSPOUSE05_running   index_empl_svy_0            0 TEMP_X_a~ -0.08~ 0.290
##  76 INCSPOUSE05_running   index_empl_svy_0            0 INT_trea~ 0.120~ 0.121
##  77 INCSPOUSE05_running   index_empl_svy_0            0 TEMP_ind~ 0.026~ 0.789
##  78 INCSPOUSE05_running   index_empl_svy_0            0 TEMP_X_a~ -0.08~ 0.535
##  79 INCSPOUSE05_running   index_empl_svy_1            0 INT_trea~ 0.121~ 0.116
##  80 INCSPOUSE05_running   index_empl_svy_1            0 TEMP_ind~ 0.010~ 0.837
##  81 INCSPOUSE05_running   index_empl_svy_1            0 TEMP_X_a~ -0.04~ 0.588
##  82 INCSPOUSE05_running   index_empl_svy_1            0 INT_trea~ 0.189~ 0.0137
##  83 INCSPOUSE05_running   index_empl_svy_1            0 TEMP_ind~ 0.027~ 0.592
##  84 INCSPOUSE05_running   index_empl_svy_1            0 TEMP_X_a~ -0.05~ 0.453
##  85 INCSPOUSE05_running   index_empl_svy_2            1 INT_trea~ 1.246~ 0.316
##  86 INCSPOUSE05_running   index_empl_svy_2            1 TEMP_ind~ 0.337~ 0.661
##  87 INCSPOUSE05_running   index_empl_svy_2            1 TEMP_X_a~ -0.14~ 0.913
##  88 INCSPOUSE05_running   index_empl_svy_2            1 INT_trea~ 0.781~ 0.524
##  89 INCSPOUSE05_running   index_empl_svy_2            1 TEMP_ind~ 0.015~ 0.992
##  90 INCSPOUSE05_running   index_empl_svy_2            1 TEMP_X_a~ -1.23~ 0.578
##  91 INCSPOUSE05_running   index_empl_svy_3            1 INT_trea~ 0.866~ 0.476
##  92 INCSPOUSE05_running   index_empl_svy_3            1 TEMP_ind~ 0.043~ 0.956
##  93 INCSPOUSE05_running   index_empl_svy_3            1 TEMP_X_a~ -0.37~ 0.754
##  94 INCSPOUSE05_running   index_empl_svy_3            1 INT_trea~ 1.703~ 0.125
##  95 INCSPOUSE05_running   index_empl_svy_3            1 TEMP_ind~ -0.43~ 0.554
##  96 INCSPOUSE05_running   index_empl_svy_3            1 TEMP_X_a~ 1.093~ 0.337
##  97 INCSPOUSE05_voteshare index_empl_svy_0            0 INT_trea~ 0.046~ 0.171
##  98 INCSPOUSE05_voteshare index_empl_svy_0            0 TEMP_ind~ -0.02~ 0.344
##  99 INCSPOUSE05_voteshare index_empl_svy_0            0 TEMP_X_a~ 0.065~ 0.0716
## 100 INCSPOUSE05_voteshare index_empl_svy_0            0 INT_trea~ 0.046~ 0.169
## 101 INCSPOUSE05_voteshare index_empl_svy_0            0 TEMP_ind~ -0.02~ 0.543
## 102 INCSPOUSE05_voteshare index_empl_svy_0            0 TEMP_X_a~ 0.055~ 0.370
## 103 INCSPOUSE05_voteshare index_empl_svy_1            0 INT_trea~ 0.047~ 0.154
## 104 INCSPOUSE05_voteshare index_empl_svy_1            0 TEMP_ind~ -0.01~ 0.578
## 105 INCSPOUSE05_voteshare index_empl_svy_1            0 TEMP_X_a~ 0.039~ 0.244
## 106 INCSPOUSE05_voteshare index_empl_svy_1            0 INT_trea~ 0.046~ 0.190
## 107 INCSPOUSE05_voteshare index_empl_svy_1            0 TEMP_ind~ -0.02~ 0.394
## 108 INCSPOUSE05_voteshare index_empl_svy_1            0 TEMP_X_a~ 0.065~ 0.0756
## 109 INCSPOUSE05_voteshare index_empl_svy_2            1 INT_trea~ 0.019~ 0.868
## 110 INCSPOUSE05_voteshare index_empl_svy_2            1 TEMP_ind~ -0.00~ 0.925
## 111 INCSPOUSE05_voteshare index_empl_svy_2            1 TEMP_X_a~ -0.08~ 0.537
## 112 INCSPOUSE05_voteshare index_empl_svy_2            1 INT_trea~ 0.025~ 0.827
## 113 INCSPOUSE05_voteshare index_empl_svy_2            1 TEMP_ind~ 0.099~ 0.637
## 114 INCSPOUSE05_voteshare index_empl_svy_2            1 TEMP_X_a~ -0.09~ 0.717
## 115 INCSPOUSE05_voteshare index_empl_svy_3            1 INT_trea~ 0.036~ 0.756
## 116 INCSPOUSE05_voteshare index_empl_svy_3            1 TEMP_ind~ -0.01~ 0.888
## 117 INCSPOUSE05_voteshare index_empl_svy_3            1 TEMP_X_a~ 0.016~ 0.897
## 118 INCSPOUSE05_voteshare index_empl_svy_3            1 INT_trea~ 0.068~ 0.502
## 119 INCSPOUSE05_voteshare index_empl_svy_3            1 TEMP_ind~ 0.002~ 0.980
## 120 INCSPOUSE05_voteshare index_empl_svy_3            1 TEMP_X_a~ -0.21~ 0.0692
## 121 INCSPOUSE05_won       index_empl_svy_0            0 INT_trea~ 1.509~ 0.596
## 122 INCSPOUSE05_won       index_empl_svy_0            0 TEMP_ind~ -0.51~ 0.836
## 123 INCSPOUSE05_won       index_empl_svy_0            0 TEMP_X_a~ -1.90~ 0.591
## 124 INCSPOUSE05_won       index_empl_svy_0            0 INT_trea~ 1.205~ 0.650
## 125 INCSPOUSE05_won       index_empl_svy_0            0 TEMP_ind~ 0.763~ 0.874
## 126 INCSPOUSE05_won       index_empl_svy_0            0 TEMP_X_a~ -1.67~ 0.786
## 127 INCSPOUSE05_won       index_empl_svy_1            0 INT_trea~ 1.358~ 0.607
## 128 INCSPOUSE05_won       index_empl_svy_1            0 TEMP_ind~ 0.204~ 0.926
## 129 INCSPOUSE05_won       index_empl_svy_1            0 TEMP_X_a~ -1.42~ 0.630
## 130 INCSPOUSE05_won       index_empl_svy_1            0 INT_trea~ 1.404~ 0.584
## 131 INCSPOUSE05_won       index_empl_svy_1            0 TEMP_ind~ -0.30~ 0.891
## 132 INCSPOUSE05_won       index_empl_svy_1            0 TEMP_X_a~ -2.52~ 0.403
## 133 INCSPOUSE05_won       index_empl_svy_2            1 INT_trea~ -0.01~ 0.820
## 134 INCSPOUSE05_won       index_empl_svy_2            1 TEMP_ind~ 0.038~ 0.462
## 135 INCSPOUSE05_won       index_empl_svy_2            1 TEMP_X_a~ -0.05~ 0.493
## 136 INCSPOUSE05_won       index_empl_svy_2            1 INT_trea~ -0.02~ 0.685
## 137 INCSPOUSE05_won       index_empl_svy_2            1 TEMP_ind~ 0.069~ 0.484
## 138 INCSPOUSE05_won       index_empl_svy_2            1 TEMP_X_a~ 0.003~ 0.982
## 139 INCSPOUSE05_won       index_empl_svy_3            1 INT_trea~ -0.01~ 0.825
## 140 INCSPOUSE05_won       index_empl_svy_3            1 TEMP_ind~ 0.021~ 0.646
## 141 INCSPOUSE05_won       index_empl_svy_3            1 TEMP_X_a~ -0.01~ 0.777
## 142 INCSPOUSE05_won       index_empl_svy_3            1 INT_trea~ -0.01~ 0.832
## 143 INCSPOUSE05_won       index_empl_svy_3            1 TEMP_ind~ 0.048~ 0.304
## 144 INCSPOUSE05_won       index_empl_svy_3            1 TEMP_X_a~ -0.07~ 0.234
## 145 INCOTHER05_running    index_empl_svy_0            0 INT_trea~ -0.30~ 0.0277
## 146 INCOTHER05_running    index_empl_svy_0            0 TEMP_ind~ 0.152~ 0.197
## 147 INCOTHER05_running    index_empl_svy_0            0 TEMP_X_a~ -0.06~ 0.702
## 148 INCOTHER05_running    index_empl_svy_0            0 INT_trea~ -0.18~ 0.141
## 149 INCOTHER05_running    index_empl_svy_0            0 TEMP_ind~ 0.039~ 0.862
## 150 INCOTHER05_running    index_empl_svy_0            0 TEMP_X_a~ -0.02~ 0.923
## 151 INCOTHER05_running    index_empl_svy_1            0 INT_trea~ -0.19~ 0.120
## 152 INCOTHER05_running    index_empl_svy_1            0 TEMP_ind~ 0.014~ 0.896
## 153 INCOTHER05_running    index_empl_svy_1            0 TEMP_X_a~ 0.043~ 0.755
## 154 INCOTHER05_running    index_empl_svy_1            0 INT_trea~ -0.19~ 0.105
## 155 INCOTHER05_running    index_empl_svy_1            0 TEMP_ind~ 0.182~ 0.0831
## 156 INCOTHER05_running    index_empl_svy_1            0 TEMP_X_a~ -0.22~ 0.107
## 157 INCOTHER05_running    index_empl_svy_2            1 INT_trea~ -5.90~ 0.104
## 158 INCOTHER05_running    index_empl_svy_2            1 TEMP_ind~ 4.593~ 0.147
## 159 INCOTHER05_running    index_empl_svy_2            1 TEMP_X_a~ -3.70~ 0.407
## 160 INCOTHER05_running    index_empl_svy_2            1 INT_trea~ -4.41~ 0.180
## 161 INCOTHER05_running    index_empl_svy_2            1 TEMP_ind~ 6.133~ 0.305
## 162 INCOTHER05_running    index_empl_svy_2            1 TEMP_X_a~ -6.12~ 0.423
## 163 INCOTHER05_running    index_empl_svy_3            1 INT_trea~ -4.15~ 0.210
## 164 INCOTHER05_running    index_empl_svy_3            1 TEMP_ind~ 0.674~ 0.806
## 165 INCOTHER05_running    index_empl_svy_3            1 TEMP_X_a~ -0.28~ 0.939
## 166 INCOTHER05_running    index_empl_svy_3            1 INT_trea~ -4.51~ 0.158
## 167 INCOTHER05_running    index_empl_svy_3            1 TEMP_ind~ 4.982~ 0.0728
## 168 INCOTHER05_running    index_empl_svy_3            1 TEMP_X_a~ -5.15~ 0.169
## 169 INCOTHER05_voteshare  index_empl_svy_0            0 INT_trea~ 0.015~ 0.730
## 170 INCOTHER05_voteshare  index_empl_svy_0            0 TEMP_ind~ -0.00~ 0.950
## 171 INCOTHER05_voteshare  index_empl_svy_0            0 TEMP_X_a~ 0.021~ 0.694
## 172 INCOTHER05_voteshare  index_empl_svy_0            0 INT_trea~ 0.020~ 0.610
## 173 INCOTHER05_voteshare  index_empl_svy_0            0 TEMP_ind~ -0.01~ 0.861
## 174 INCOTHER05_voteshare  index_empl_svy_0            0 TEMP_X_a~ 0.041~ 0.652
## 175 INCOTHER05_voteshare  index_empl_svy_1            0 INT_trea~ 0.020~ 0.605
## 176 INCOTHER05_voteshare  index_empl_svy_1            0 TEMP_ind~ -0.00~ 0.800
## 177 INCOTHER05_voteshare  index_empl_svy_1            0 TEMP_X_a~ 0.031~ 0.478
## 178 INCOTHER05_voteshare  index_empl_svy_1            0 INT_trea~ 0.025~ 0.528
## 179 INCOTHER05_voteshare  index_empl_svy_1            0 TEMP_ind~ 0.001~ 0.987
## 180 INCOTHER05_voteshare  index_empl_svy_1            0 TEMP_X_a~ 0.010~ 0.824
## 181 INCOTHER05_voteshare  index_empl_svy_2            1 INT_trea~ 0.050~ 0.500
## 182 INCOTHER05_voteshare  index_empl_svy_2            1 TEMP_ind~ -0.02~ 0.668
## 183 INCOTHER05_voteshare  index_empl_svy_2            1 TEMP_X_a~ 0.214~ 0.0237
## 184 INCOTHER05_voteshare  index_empl_svy_2            1 INT_trea~ 0.093~ 0.181
## 185 INCOTHER05_voteshare  index_empl_svy_2            1 TEMP_ind~ -0.04~ 0.717
## 186 INCOTHER05_voteshare  index_empl_svy_2            1 TEMP_X_a~ 0.343~ 0.0371
## 187 INCOTHER05_voteshare  index_empl_svy_3            1 INT_trea~ 0.109~ 0.117
## 188 INCOTHER05_voteshare  index_empl_svy_3            1 TEMP_ind~ -0.04~ 0.480
## 189 INCOTHER05_voteshare  index_empl_svy_3            1 TEMP_X_a~ 0.188~ 0.0166
## 190 INCOTHER05_voteshare  index_empl_svy_3            1 INT_trea~ 0.090~ 0.204
## 191 INCOTHER05_voteshare  index_empl_svy_3            1 TEMP_ind~ -0.01~ 0.863
## 192 INCOTHER05_voteshare  index_empl_svy_3            1 TEMP_X_a~ 0.109~ 0.186
## 193 INCOTHER05_won        index_empl_svy_0            0 INT_trea~ 1.661~ 0.529
## 194 INCOTHER05_won        index_empl_svy_0            0 TEMP_ind~ -0.16~ 0.942
## 195 INCOTHER05_won        index_empl_svy_0            0 TEMP_X_a~ 3.232~ 0.327
## 196 INCOTHER05_won        index_empl_svy_0            0 INT_trea~ 2.922~ 0.233
## 197 INCOTHER05_won        index_empl_svy_0            0 TEMP_ind~ -1.67~ 0.705
## 198 INCOTHER05_won        index_empl_svy_0            0 TEMP_X_a~ 7.116~ 0.213
## 199 INCOTHER05_won        index_empl_svy_1            0 INT_trea~ 3.426~ 0.165
## 200 INCOTHER05_won        index_empl_svy_1            0 TEMP_ind~ -1.37~ 0.503
## 201 INCOTHER05_won        index_empl_svy_1            0 TEMP_X_a~ 3.275~ 0.235
## 202 INCOTHER05_won        index_empl_svy_1            0 INT_trea~ 3.444~ 0.167
## 203 INCOTHER05_won        index_empl_svy_1            0 TEMP_ind~ 0.282~ 0.895
## 204 INCOTHER05_won        index_empl_svy_1            0 TEMP_X_a~ 0.436~ 0.880
## 205 INCOTHER05_won        index_empl_svy_2            1 INT_trea~ 0.027~ 0.515
## 206 INCOTHER05_won        index_empl_svy_2            1 TEMP_ind~ 0.001~ 0.981
## 207 INCOTHER05_won        index_empl_svy_2            1 TEMP_X_a~ 0.003~ 0.957
## 208 INCOTHER05_won        index_empl_svy_2            1 INT_trea~ 0.039~ 0.318
## 209 INCOTHER05_won        index_empl_svy_2            1 TEMP_ind~ -0.02~ 0.703
## 210 INCOTHER05_won        index_empl_svy_2            1 TEMP_X_a~ 0.038~ 0.676
## 211 INCOTHER05_won        index_empl_svy_3            1 INT_trea~ 0.044~ 0.249
## 212 INCOTHER05_won        index_empl_svy_3            1 TEMP_ind~ -0.02~ 0.523
## 213 INCOTHER05_won        index_empl_svy_3            1 TEMP_X_a~ 0.010~ 0.810
## 214 INCOTHER05_won        index_empl_svy_3            1 INT_trea~ 0.042~ 0.273
## 215 INCOTHER05_won        index_empl_svy_3            1 TEMP_ind~ 0.006~ 0.852
## 216 INCOTHER05_won        index_empl_svy_3            1 TEMP_X_a~ -0.01~ 0.785
```

```r
# Comparing selected controls and baseline controls
baseline_controls <- gpcontrols
selected_controls_list <- map(models_list, ~ {
  vars <- names(coef(.x))[-1]
  setdiff(vars, treat_vars)
})

controls_diff <- tibble::tibble(
  dep_var = rep(incum_dep_vars2, each = length(indices) * 2),
  index = rep(rep(indices, each = 2), length(incum_dep_vars2)),
  RES05_gender = rep(rep(c(0, 1), each = length(indices)), length(incum_dep_vars2)),
  selected = selected_controls_list
) %>%
  mutate(
    kept = map(selected, ~ intersect(.x, baseline_controls)),
    dropped = map(selected, ~ setdiff(baseline_controls, .x)),
    added = map(selected, ~ setdiff(.x, baseline_controls)),
    n_selected = map_int(selected, length)
  )

cat("\n==== COMPARING CONTROLS (baseline vs double lasso) ====\n")
```

```
##
## ==== COMPARING CONTROLS (baseline vs double lasso) ====
```

```r
for (r in 1:nrow(controls_diff)) {
  cat("\n--- DV:", controls_diff$dep_var[r],
      " | Index:", controls_diff$index[r],
      " | RES05_gender:", controls_diff$RES05_gender[r], "--- \n")
  cat("# Selected controls =", controls_diff$n_selected[r], "\n")
  cat("Kept (baseline)   :", paste(controls_diff$kept[[r]], collapse = ", "), "\n")
  cat("Dropped (baseline):", paste(controls_diff$dropped[[r]], collapse = ", "), "\n")
  cat("Added (non-base)  :", paste(controls_diff$added[[r]], collapse = ", "), "\n")
}
```

```
##
## --- DV: INC05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 15
## Kept (baseline)   : GP_population, RES00_obc, RES10_obc, RES10_st, RES05_sc, RES00_gender, RES00_sc,
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES10_sc, RES05_obc, RES05_st
## Added (non-base)  : CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_BHARATPUR, fe_DHOLPUR, fe_
##
## --- DV: INC05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 18
## Kept (baseline)   : GP_population, RES00_gender, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_st,
## Dropped (baseline): GP_lit, GP_st, GP_nbvillages, RES10_sc, RES05_obc, RES05_st
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_BHA
##
## --- DV: INC05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 15
## Kept (baseline)   : GP_population, RES00_gender, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_st,
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES10_sc, RES05_obc, RES05_st
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_BHA
##
## --- DV: INC05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 21
## Kept (baseline)   : GP_population, RES00_obc, RES00_sc, RES10_obc, RES10_st, RES05_sc, RES00_gender,
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc
## Added (non-base)  : CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_BHARATPUR, fe_DHOLPUR, fe_
##
## --- DV: INC05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_powe
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_powe
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_powe
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_powe
##
## --- DV: INC05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_powe
##
## --- DV: INC05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INC05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_cult, CENSUS_VD2001_pow
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES10_st, R
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 12
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_sc, RES1
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 9
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_obc, RES10_obc, RES10_
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INCSPOUSE05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 20
## Kept (baseline)   : RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc, GP_st, RES00_obc, RES10_obc, RES
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc, RES05_sc
## Added (non-base)  : CENSUS_PCA2001_tot_cult, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_B
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 15
## Kept (baseline)   : RES00_obc, RES10_obc, RES05_sc, RES00_gender, RES00_sc, RES00_st, GP_population
## Dropped (baseline): GP_lit, GP_sc, GP_st, GP_nbvillages, RES10_sc, RES10_st, RES05_obc, RES05_st
## Added (non-base)  : CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 15
## Kept (baseline)   : RES00_obc, RES10_obc, RES05_sc, RES00_gender, RES00_sc, RES00_st, GP_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES10_sc, RES10_st, RES05_obc, RES0
## Added (non-base)  : CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 13
## Kept (baseline)   : RES00_obc, RES10_obc, RES05_sc, RES00_gender, RES00_sc, RES00_st
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES10_sc, RES10_st, RES05_ob
## Added (non-base)  : CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 21
## Kept (baseline)   : RES00_obc, RES10_obc, RES05_sc, RES00_gender, RES00_sc, RES00_st, GP_lit, GP_sc,
## Dropped (baseline): GP_nbvillages, RES10_sc, RES05_obc
## Added (non-base)  : CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 13
## Kept (baseline)   : GP_sc, RES00_obc, RES05_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES00_st, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power
##
## --- DV: INCSPOUSE05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 8
## Kept (baseline)   : GP_sc, RES05_sc
## Dropped (baseline): GP_population, GP_lit, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES00_sc, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power
##
## --- DV: INCSPOUSE05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_running  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 12
## Kept (baseline)   : GP_lit, RES00_obc
## Dropped (baseline): GP_population, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RE
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_voteshare  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 7
## Kept (baseline)   : GP_lit
## Dropped (baseline): GP_population, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES00_sc, R
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_powe
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_0  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_1  | RES05_gender: 0 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 11
## Kept (baseline)   : RES00_obc
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES00
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_2  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 5
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
##
## --- DV: INCOTHER05_won  | Index: index_empl_svy_3  | RES05_gender: 1 ---
## # Selected controls = 6
## Kept (baseline)   :
## Dropped (baseline): GP_population, GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES0
## Added (non-base)  : CENSUS_PCA2001_tot_lit, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KA
```

```r
cat("\n==== END ====\n")
```

```
##
## ==== END ====
```

Replication of Table 3 using double lasso.

```r
#################################### TABLE 3 #####################################

## DATA AND MACROS ##

# Baseline list of controls for table 3
gpcontrols_table3 <- c("GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
                       "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
                       "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st")

# Full list of candidates
full_controls_candidates <- c(
  # Baseline elements:
  "GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
  "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
  "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st",

  # Added from the census:
  "CENSUS_PCA2001_tot_pop", "CENSUS_PCA2001_tot_lit", "CENSUS_PCA2001_tot_sc",
  "CENSUS_PCA2001_tot_st", "CENSUS_PCA2001_tot_aglb", "CENSUS_PCA2001_tot_nm_hh",
  "CENSUS_PCA2001_tot_cult", "CENSUS_VD2001_power_dom", "CENSUS_VD2001_drnk_wat_f",
  "CENSUS_VD2001_edu_fac", "CENSUS_VD2001_medi_fac"
)

# Dependent variables
dep_vars_table3 <- c("ELEC10_nbcands", "CHAL_nbchal", "CHAL_prop_female",
                     "CHAL_voteshare_female", "CHAL_prop_nongen", "CHAL_voteshare_nongen")

# Treatment variables
outregvar2 <- c("INT_treatment", "RES05_gender", "X_anytr_genderres05")

# Loading and filtering the data
data <- haven::read_dta("~/work/Electoral data cleaned.dta")
data_filtered <- data %>% filter(RES10_gender == 0, GP_tag == 1)



## FUNCTIONS ##

# Non zero coefficients
get_nonzero_names <- function(cf, drop = "(Intercept)") {
  cf_mat <- as.matrix(cf)
  nz <- which(cf_mat != 0)
  vars <- rownames(cf_mat)[nz]
  setdiff(vars, drop)
}

# Double lasso function

double_lasso_fit <- function(data_df, y_name, d_names, x_candidates, baseline_names = NULL, fe_name = "
  # FE
  if (!is.null(fe_name)) {
    fe_vec <- data_df[[fe_name]]
    fe_mat <- model.matrix(~ factor(fe_vec) - 1)
    colnames(fe_mat) <- paste0("fe_", levels(factor(fe_vec)))
  } else {
    fe_mat <- NULL
  }

  # X : candidates
  X_all <- data_df %>% dplyr::select(all_of(x_candidates))
```

```
X_mat <- model.matrix(~ . - 1, data = X_all)

# Filter NA rows
keep_rows_X <- complete.cases(X_mat)
if (!is.null(fe_mat)) {
  keep_rows_FE <- complete.cases(fe_mat)
  keep_rows <- keep_rows_X & keep_rows_FE
  X_mat <- X_mat[keep_rows, , drop = FALSE]
  fe_mat <- fe_mat[keep_rows, , drop = FALSE]
} else {
  X_mat <- X_mat[keep_rows_X, , drop = FALSE]
  keep_rows <- keep_rows_X
}

baseline_idx <- if (!is.null(baseline_names)) {
  which(colnames(X_mat) %in% baseline_names)
} else integer(0)

if (!is.null(fe_mat)) {
  X_mat <- cbind(X_mat, fe_mat)
  fe_idx <- (ncol(X_mat) - ncol(fe_mat) + 1):ncol(X_mat)
} else {
  fe_idx <- integer(0)
}

# Y and D
missing_d <- setdiff(d_names, names(data_df))
if (length(missing_d) > 0) {
  for (nm in missing_d) {
    parts <- strsplit(nm, ":", fixed = TRUE)[[1]]
    stopifnot(all(parts %in% names(data_df)))
    data_df[[nm]] <- data_df[[parts[1]]] * data_df[[parts[2]]]
  }
}

y_vec <- as.numeric(data_df[[y_name]])
D_mat <- as.matrix(data_df %>% dplyr::select(all_of(d_names)))
if (is.null(ncol(D_mat))) D_mat <- matrix(D_mat, ncol = 1)

# Filter y_vec and D_mat
y_vec <- y_vec[keep_rows]
D_mat <- D_mat[keep_rows, , drop = FALSE]
if (!is.null(se_cluster)) se_cluster <- se_cluster[keep_rows]

# Checks
y_constant <- length(unique(y_vec)) < 2
d_constant <- apply(D_mat, 2, function(z) length(unique(z)) < 2)

# Y ~ D + X
sel_y <- character(0); lambda_y <- NA
if (!y_constant) {
  X_y <- cbind(D_mat, X_mat)
  pf_y <- c(rep(0, ncol(D_mat)), rep(1, ncol(X_mat)))
  if (length(fe_idx) > 0) pf_y[ncol(D_mat) + fe_idx] <- 0

  if (all(pf_y == 0)) {
    sel_y <- character(0)
    lambda_y <- NA
  } else {
    set.seed(123)
    fit1 <- tryCatch({
      cv.glmnet(X_y, y_vec, alpha = 1, family = family, penalty.factor = pf_y)
    }, error = function(e) NULL)

    if (!is.null(fit1)) {
      coef1 <- coef(fit1, s = "lambda.min")
      cf_mat1 <- as.matrix(coef1)
      nz1 <- which(cf_mat1 != 0)
      sel_y <- rownames(cf_mat1)[nz1]
      sel_y <- setdiff(sel_y, c("(Intercept)", colnames(D_mat)))
      lambda_y <- fit1$lambda.min
    } else {
      sel_y <- character(0)
      lambda_y <- NA
    }
  }
}

# D_k ~ X
pf_d <- rep(1, ncol(X_mat))
if (length(baseline_idx) > 0) pf_d[baseline_idx] <- 1
if (length(fe_idx) > 0) pf_d[fe_idx] <- 0

sel_d <- character(0)
d_lambdas <- rep(NA, ncol(D_mat))

if (!all(pf_d == 0)) {
  for (k in seq_len(ncol(D_mat))) {
    if (d_constant[k]) next
    set.seed(123)
    fitk <- tryCatch({
      cv.glmnet(X_mat, D_mat[, k], alpha = 1, family = "gaussian", penalty.factor = pf_d)
    }, error = function(e) NULL)

    if (!is.null(fitk)) {
      coefk <- coef(fitk, s = "lambda.min")
      cf_matk <- as.matrix(coefk)
      nz_k <- which(cf_matk != 0)
      sel_k <- rownames(cf_matk)[nz_k]
      sel_k <- setdiff(sel_k, "(Intercept)")
      sel_d <- union(sel_d, sel_k)
      d_lambdas[k] <- fitk$lambda.min
    }
  }
}
```

```r
  # Union and final regression
  selected_controls <- union(sel_y, sel_d)
  if (length(fe_idx) > 0) selected_controls <- union(selected_controls, colnames(X_mat)[fe_idx])

  X_final <- X_mat[, selected_controls, drop = FALSE]
  final_df <- data.frame(Y = y_vec, D_mat, X_final)
  colnames(final_df)[2:(1 + ncol(D_mat))] <- colnames(D_mat)

  rhs <- paste(c(colnames(D_mat), colnames(X_final)), collapse = " + ")
  f_final <- as.formula(paste("Y ~", rhs))
  lm_final <- lm(f_final, data = final_df)

  vc <- if (!is.null(se_cluster)) {
    sandwich::vcovCL(lm_final, cluster = se_cluster)
  } else {
    sandwich::vcovHC(lm_final, type = "HC1")
  }

  coefs_rob <- lmtest::coeftest(lm_final, vcov = vc)

  list(
    selected_controls = selected_controls,
    model = lm_final,
    robust_coefs = coefs_rob,
    lambda_y = lambda_y,
    d_lambdas = d_lambdas,
    y_constant = y_constant,
    d_constant = d_constant
  )
}

# Checking missing data among candidates
missing_data <- data_filtered %>% select(all_of(full_controls_candidates)) %>% summarise_all(~sum(is.na
print(missing_data, n = Inf, width = Inf)
```

```
## # A tibble: 1 x 26
##   GP_population GP_lit GP_sc GP_st GP_nbvillages RES00_gender RES00_obc RES00_sc
##           <int>  <int> <int> <int>         <int>        <int>     <int>    <int>
## 1             0      0     0     0             0            0         0        0
##   RES00_st RES10_obc RES10_sc RES10_st RES05_obc RES05_sc RES05_st
##      <int>     <int>    <int>    <int>     <int>    <int>    <int>
## 1        0         0        0        0         0        0        0
##   CENSUS_PCA2001_tot_pop CENSUS_PCA2001_tot_lit CENSUS_PCA2001_tot_sc
##                    <int>                  <int>                 <int>
## 1                      0                      0                     0
##   CENSUS_PCA2001_tot_st CENSUS_PCA2001_tot_aglb CENSUS_PCA2001_tot_nm_hh
##                   <int>                   <int>                    <int>
## 1                     0                       0                        0
##   CENSUS_PCA2001_tot_cult CENSUS_VD2001_power_dom CENSUS_VD2001_drnk_wat_f
##                     <int>                   <int>                    <int>
## 1                       0                       0                        0
##   CENSUS_VD2001_edu_fac CENSUS_VD2001_medi_fac
##                   <int>                  <int>
## 1                     0                      0
```

```
# Function for Wald tests
calculate_tests <- function(model) {
  pval1 <- tryCatch(car::linearHypothesis(model, "RES05_gender = 0")$`Pr(>F)`[2], error = function(e) N
  pval2 <- tryCatch(car::linearHypothesis(model, "X_anytr_genderres05 = 0")$`Pr(>F)`[2], error = functi
  pval3 <- tryCatch(car::linearHypothesis(model, "INT_treatment + X_anytr_genderres05 = 0")$`Pr(>F)`[2]

  return(list(pval1 = round(pval1, 2), pval2 = round(pval2, 2), pval3 = round(pval3, 2)))
}


## LOOP FOR REGRESSIONS ##

models_list <- list()
control_means <- list()
test_results <- list()

# For each subsample
subsets <- list(
  list(data = data_filtered, name = "Full Sample"),
  list(data = data_filtered %>% filter(INC05_can_run == 1), name = "Incumbent Can Run"),
  list(data = data_filtered %>% filter(INC05_can_run == 0), name = "Incumbent Cannot Run")
)

for (subset in subsets) {
  data_subset <- subset$data
  subset_name <- subset$name
  cat(paste("\n=== ANALYZE FOR:", subset_name, "===\n"))
  cat("Number of observations:", nrow(data_subset), "\n")

  # Filter NA data
  required_vars <- c(full_controls_candidates, "district", dep_vars_table3, outregvar2)
  data_subset <- data_subset[, required_vars, drop = FALSE]
  data_subset <- na.omit(data_subset)

  for (i in seq_along(dep_vars_table3)) {
    dep_var <- dep_vars_table3[i]
    cat(paste("Dependent variable:", dep_var, "\n"))

    if (!dep_var %in% names(data_subset)) {
      cat(paste("    ATTENTION: Variable", dep_var, "not found!\n"))
      next
    }

    # Control mean
    control_mean <- data_subset %>%
      filter(INT_treatment == 0, RES05_gender == 0) %>%
      summarise(mean_val = mean(!!sym(dep_var), na.rm = TRUE)) %>%
      pull(mean_val) %>%
      round(2)
    control_means[[i]] <- control_mean
    cat(paste("    Control mean (non-previously gender-reserved):", control_mean, "\n"))

    #  Treatment variables
    d_names <- outregvar2

    # Double lasso
    results <- double_lasso_fit(
      data_df = data_subset,
      y_name = dep_var,
      d_names = d_names,
      x_candidates = full_controls_candidates,
      baseline_names = gpcontrols_table3,
      fe_name = "district",
      se_cluster = data_subset$district
    )

    models_list[[paste0(subset_name, "_", dep_var)]] <- results$model
    test_results[[paste0(subset_name, "_", dep_var)]] <- calculate_tests(results$model)
  }
}
```

```
##
## === ANALYZE FOR: Full Sample ===
## Number of observations: 382
## Dependent variable: ELEC10_nbcands
##     Control mean (non-previously gender-reserved): 7.5
## Dependent variable: CHAL_nbchal
##     Control mean (non-previously gender-reserved): 7.24
## Dependent variable: CHAL_prop_female
##     Control mean (non-previously gender-reserved): 0.12
## Dependent variable: CHAL_voteshare_female
##     Control mean (non-previously gender-reserved): 12.89
## Dependent variable: CHAL_prop_nongen
##     Control mean (non-previously gender-reserved): 0.78
## Dependent variable: CHAL_voteshare_nongen
##     Control mean (non-previously gender-reserved): 78.7
##
## === ANALYZE FOR: Incumbent Can Run ===
## Number of observations: 245
## Dependent variable: ELEC10_nbcands
##     Control mean (non-previously gender-reserved): 7.53
## Dependent variable: CHAL_nbchal
##     Control mean (non-previously gender-reserved): 7.12
## Dependent variable: CHAL_prop_female
##     Control mean (non-previously gender-reserved): 0.1
## Dependent variable: CHAL_voteshare_female
##     Control mean (non-previously gender-reserved): 10.74
## Dependent variable: CHAL_prop_nongen
##     Control mean (non-previously gender-reserved): 0.65
## Dependent variable: CHAL_voteshare_nongen
##     Control mean (non-previously gender-reserved): 66.04
##
## === ANALYZE FOR: Incumbent Cannot Run ===
## Number of observations: 137
## Dependent variable: ELEC10_nbcands
##     Control mean (non-previously gender-reserved): 7.45
## Dependent variable: CHAL_nbchal
##     Control mean (non-previously gender-reserved): 7.45
## Dependent variable: CHAL_prop_female
##     Control mean (non-previously gender-reserved): 0.16
## Dependent variable: CHAL_voteshare_female
##     Control mean (non-previously gender-reserved): 16.51
## Dependent variable: CHAL_prop_nongen
##     Control mean (non-previously gender-reserved): 1
## Dependent variable: CHAL_voteshare_nongen
##     Control mean (non-previously gender-reserved): 100
```

```
## OUTPUT ##

# Table of treatment variables' coefficients
tidy_list <- map(models_list, ~ broom::tidy(.x))
tidy_list <- Map(function(df, id) { df$model_id <- id; df }, tidy_list, names(models_list))
treat_table <- bind_rows(tidy_list) %>%
  filter(term %in% outregvar2) %>%
  mutate(
    dep_var = gsub("_[^_]+$", "", model_id),
    subset = gsub("^.*_", "", model_id),
    est_se = sprintf("%.3f (%.3f)", estimate, std.error)
  ) %>%
  select(dep_var, subset, term, est_se, p.value)

cat("==== TREATMENT: COEFS AND SE ====\n")
```

```
## ==== TREATMENT: COEFS AND SE ====
```

```
print(treat_table, row.names = FALSE, n=Inf)
```

```
## # A tibble: 54 x 5
##    dep_var                             subset  term               est_se p.value
##    <chr>                               <chr>   <chr>              <chr>    <dbl>
##  1 Full Sample_ELEC10                  nbcands INT_treatment      0.466~  0.406
##  2 Full Sample_ELEC10                  nbcands RES05_gender       1.009~  0.0781
##  3 Full Sample_ELEC10                  nbcands X_anytr_genderres~ -1.10~  0.238
##  4 Full Sample_CHAL                    nbchal  INT_treatment      0.552~  0.329
##  5 Full Sample_CHAL                    nbchal  RES05_gender       1.186~  0.0401
##  6 Full Sample_CHAL                    nbchal  X_anytr_genderres~ -1.21~  0.197
##  7 Full Sample_CHAL_prop               female  INT_treatment      -0.00~  0.761
##  8 Full Sample_CHAL_prop               female  RES05_gender       0.020~  0.453
##  9 Full Sample_CHAL_prop               female  X_anytr_genderres~ 0.018~  0.674
## 10 Full Sample_CHAL_voteshare          female  INT_treatment      -2.83~  0.339
## 11 Full Sample_CHAL_voteshare          female  RES05_gender       4.263~  0.160
## 12 Full Sample_CHAL_voteshare          female  X_anytr_genderres~ 0.884~  0.858
## 13 Full Sample_CHAL_prop               nongen  INT_treatment      0.068~  0.0474
## 14 Full Sample_CHAL_prop               nongen  RES05_gender       0.052~  0.140
## 15 Full Sample_CHAL_prop               nongen  X_anytr_genderres~ -0.05~  0.324
## 16 Full Sample_CHAL_voteshare          nongen  INT_treatment      7.602~  0.0330
## 17 Full Sample_CHAL_voteshare          nongen  RES05_gender       4.498~  0.219
## 18 Full Sample_CHAL_voteshare          nongen  X_anytr_genderres~ -7.86~  0.185
## 19 Incumbent Can Run_ELEC10            nbcands INT_treatment      0.868~  0.256
## 20 Incumbent Can Run_ELEC10            nbcands RES05_gender       1.161~  0.0919
## 21 Incumbent Can Run_ELEC10            nbcands X_anytr_genderres~ -1.70~  0.133
## 22 Incumbent Can Run_CHAL              nbchal  INT_treatment      1.103~  0.154
## 23 Incumbent Can Run_CHAL              nbchal  RES05_gender       1.488~  0.0330
## 24 Incumbent Can Run_CHAL              nbchal  X_anytr_genderres~ -2.05~  0.0763
## 25 Incumbent Can Run_CHAL_prop         female  INT_treatment      0.052~  0.0999
## 26 Incumbent Can Run_CHAL_prop         female  RES05_gender       0.009~  0.755
## 27 Incumbent Can Run_CHAL_prop         female  X_anytr_genderres~ -0.03~  0.510
## 28 Incumbent Can Run_CHAL_voteshare    female  INT_treatment      2.682~  0.469
## 29 Incumbent Can Run_CHAL_voteshare    female  RES05_gender       4.091~  0.220
## 30 Incumbent Can Run_CHAL_voteshare    female  X_anytr_genderres~ -5.24~  0.342
## 31 Incumbent Can Run_CHAL_prop         nongen  INT_treatment      0.083~  0.151
## 32 Incumbent Can Run_CHAL_prop         nongen  RES05_gender       0.087~  0.0938
## 33 Incumbent Can Run_CHAL_prop         nongen  X_anytr_genderres~ -0.06~  0.447
## 34 Incumbent Can Run_CHAL_voteshare    nongen  INT_treatment      8.788~  0.144
## 35 Incumbent Can Run_CHAL_voteshare    nongen  RES05_gender       7.980~  0.140
## 36 Incumbent Can Run_CHAL_voteshare    nongen  X_anytr_genderres~ -9.32~  0.297
## 37 Incumbent Cannot Run_ELEC10         nbcands INT_treatment      0.522~  0.561
## 38 Incumbent Cannot Run_ELEC10         nbcands RES05_gender       1.246~  0.229
## 39 Incumbent Cannot Run_ELEC10         nbcands X_anytr_genderres~ -0.41~  0.810
## 40 Incumbent Cannot Run_CHAL           nbchal  INT_treatment      0.529~  0.555
## 41 Incumbent Cannot Run_CHAL           nbchal  RES05_gender       1.213~  0.241
## 42 Incumbent Cannot Run_CHAL           nbchal  X_anytr_genderres~ -0.37~  0.828
## 43 Incumbent Cannot Run_CHAL_prop      female  INT_treatment      -0.03~  0.421
## 44 Incumbent Cannot Run_CHAL_prop      female  RES05_gender       0.040~  0.476
## 45 Incumbent Cannot Run_CHAL_prop      female  X_anytr_genderres~ 0.015~  0.869
## 46 Incumbent Cannot Run_CHAL_voteshare female  INT_treatment      -6.84~  0.207
## 47 Incumbent Cannot Run_CHAL_voteshare female  RES05_gender       3.592~  0.568
## 48 Incumbent Cannot Run_CHAL_voteshare female  X_anytr_genderres~ 4.238~  0.682
## 49 Incumbent Cannot Run_CHAL_prop      nongen  INT_treatment      0.003~  0.760
## 50 Incumbent Cannot Run_CHAL_prop      nongen  RES05_gender       -0.01~  0.140
## 51 Incumbent Cannot Run_CHAL_prop      nongen  X_anytr_genderres~ 0.012~  0.471
## 52 Incumbent Cannot Run_CHAL_voteshare nongen  INT_treatment      0.378~  0.760
## 53 Incumbent Cannot Run_CHAL_voteshare nongen  RES05_gender       -2.13~  0.140
## 54 Incumbent Cannot Run_CHAL_voteshare nongen  X_anytr_genderres~ 1.705~  0.471
```

```r
# Comparison of selected and baseline controls
baseline_controls <- gpcontrols_table3
selected_controls_list <- map(models_list, ~ {
  vars <- names(coef(.x))[-1]
  setdiff(vars, outregvar2)
})

controls_diff <- tibble::tibble(
  dep_var = rep(dep_vars_table3, length(subsets)),
  subset = rep(names(subsets), each = length(dep_vars_table3)),
  selected = selected_controls_list
) %>%
  mutate(
    kept = map(selected, ~ intersect(.x, baseline_controls)),
    dropped = map(selected, ~ setdiff(baseline_controls, .x)),
    added = map(selected, ~ setdiff(.x, baseline_controls)),
    n_selected = map_int(selected, length)
  )

cat("\n==== CONTROLS COMPARISON (baseline vs double lasso) ====\n")
```

```
##
## ==== CONTROLS COMPARISON (baseline vs double lasso) ====
```

```r
for (r in 1:nrow(controls_diff)) {
  cat("\n--- DV:", controls_diff$dep_var[r],
      " | Subset:", controls_diff$subset[r], " ---\n")
  cat("# Selected controls =", controls_diff$n_selected[r], "\n")
  cat("Kept (baseline)   :", paste(controls_diff$kept[[r]], collapse = ", "), "\n")
  cat("Dropped (baseline):", paste(controls_diff$dropped[[r]], collapse = ", "), "\n")
  cat("Added (non-base)  :", paste(controls_diff$added[[r]], collapse = ", "), "\n")
}
```

```
## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: ELEC10_nbcands  | Subset:  ---
## # Selected controls = 24
## Kept (baseline)   : GP_population, GP_lit, GP_sc, GP_nbvillages, RES00_gender, RES00_sc, RES10_obc, I
## Dropped (baseline): RES10_st
## Added (non-base)  : CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_drnk_wat_f, CENSUS_VD2001_medi_fac, fe_BHA

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_nbchal  | Subset:  ---
## # Selected controls = 25
## Kept (baseline)   : GP_population, GP_lit, GP_sc, RES00_gender, RES00_sc, RES10_obc, RES05_obc, RES05
## Dropped (baseline): RES10_st
## Added (non-base)  : CENSUS_PCA2001_tot_aglb, CENSUS_PCA2001_tot_nm_hh, CENSUS_VD2001_drnk_wat_f, CENS

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_prop_female  | Subset:  ---
## # Selected controls = 24
## Kept (baseline)   : GP_lit, GP_st, GP_nbvillages, RES00_gender, RES00_obc, RES00_sc, RES10_sc, RES10_
## Dropped (baseline): GP_population, RES05_obc
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KARA

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_voteshare_female  | Subset:  ---
## # Selected controls = 24
## Kept (baseline)   : GP_lit, GP_nbvillages, RES00_gender, RES00_obc, RES00_sc, RES10_sc, RES10_st, RES
## Dropped (baseline): GP_population, RES05_obc
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_power_dom, fe_BHARA

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_prop_nongen  | Subset:  ---
## # Selected controls = 26
## Kept (baseline)   : GP_population, GP_lit, RES00_gender, RES00_sc, RES10_obc, RES10_sc, RES10_st, RES
## Dropped (baseline): RES05_obc
## Added (non-base)  : CENSUS_PCA2001_tot_sc, CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, fe_BHARATI

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_voteshare_nongen  | Subset:  ---
## # Selected controls = 25
## Kept (baseline)   : GP_lit, GP_sc, RES00_gender, RES00_sc, RES10_obc, RES10_sc, RES10_st, RES05_sc,
## Dropped (baseline): GP_population, RES05_obc
## Added (non-base)  : CENSUS_PCA2001_tot_sc, CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, fe_BHARAT

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: ELEC10_nbcands  | Subset:  ---
## # Selected controls = 21
## Kept (baseline)   : GP_population, GP_sc, RES00_sc, RES10_obc, RES05_sc, RES05_st, RES00_gender, RES0
## Dropped (baseline): GP_lit, GP_st, GP_nbvillages, RES10_sc
## Added (non-base)  : CENSUS_PCA2001_tot_sc, CENSUS_PCA2001_tot_aglb, CENSUS_PCA2001_tot_cult, CENSUS_

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_nbchal  | Subset:  ---
## # Selected controls = 23
## Kept (baseline)   : GP_sc, GP_nbvillages, RES00_sc, RES10_obc, RES05_sc, RES05_st, GP_population, RES
## Dropped (baseline): GP_lit, GP_st, RES10_sc
## Added (non-base)  : CENSUS_PCA2001_tot_sc, CENSUS_PCA2001_tot_aglb, CENSUS_PCA2001_tot_nm_hh, CENSUS_

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_prop_female  | Subset:  ---
## # Selected controls = 22
## Kept (baseline)   : GP_lit, GP_sc, RES00_obc, RES00_sc, RES10_sc, RES10_st, RES05_sc, RES05_st, GP_p
## Dropped (baseline): GP_st, GP_nbvillages
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, CENSUS_VD2001_drnk_wat_f, fe_BHA

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_voteshare_female  | Subset:  ---
## # Selected controls = 22
## Kept (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES10_sc, RES10_st, RES05_sc,
## Dropped (baseline): GP_st, GP_nbvillages
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_PCA2001_tot_aglb, CENSUS_VD2001_power_dom, fe_BHAR

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_prop_nongen  | Subset:  ---
## # Selected controls = 22
## Kept (baseline)   : GP_lit, GP_st, RES00_gender, RES00_sc, RES10_obc, RES10_sc, RES10_st, RES05_obc,
## Dropped (baseline): GP_sc, GP_nbvillages
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KAR

## Warning: Unknown or uninitialised column: `subset`.

##
## --- DV: CHAL_voteshare_nongen  | Subset:  ---
## # Selected controls = 21
## Kept (baseline)   : GP_lit, RES00_gender, RES00_sc, RES10_obc, RES10_sc, RES10_st, RES05_sc, RES05_s
## Dropped (baseline): GP_sc, GP_st, GP_nbvillages
## Added (non-base)  : CENSUS_PCA2001_tot_st, CENSUS_VD2001_power_dom, fe_BHARATPUR, fe_DHOLPUR, fe_KAR

## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: ELEC10_nbcands  | Subset:  ---
## # Selected controls = 12
## Kept (baseline)   : GP_population, GP_sc, GP_nbvillages, RES05_sc, GP_lit, RES00_gender
## Dropped (baseline): GP_st, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc, RES10_st, RES05_obc,
## Added (non-base)  : CENSUS_VD2001_power_dom, CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KA
```

```
## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: CHAL_nbchal  | Subset:  ---
## # Selected controls = 12
## Kept (baseline)   : GP_population, GP_sc, GP_nbvillages, RES05_sc, GP_lit, RES00_gender
## Dropped (baseline): GP_st, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc, RES10_st, RES05_obc,
## Added (non-base)  : CENSUS_VD2001_power_dom, CENSUS_VD2001_medi_fac, fe_BHARATPUR, fe_DHOLPUR, fe_KA
```

```
## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: CHAL_prop_female  | Subset:  ---
## # Selected controls = 8
## Kept (baseline)   : GP_population, GP_lit, RES00_gender
## Dropped (baseline): GP_sc, GP_st, GP_nbvillages, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc,
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_medi_
```

```
## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: CHAL_voteshare_female  | Subset:  ---
## # Selected controls = 8
## Kept (baseline)   : GP_population, GP_lit, RES00_gender
## Dropped (baseline): GP_sc, GP_st, GP_nbvillages, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc,
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_medi_
```

```
## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: CHAL_prop_nongen  | Subset:  ---
## # Selected controls = 8
## Kept (baseline)   : GP_population, GP_lit, RES00_gender
## Dropped (baseline): GP_sc, GP_st, GP_nbvillages, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc,
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_medi_
```

```
## Warning: Unknown or uninitialised column: `subset`.
```

```
##
## --- DV: CHAL_voteshare_nongen  | Subset:  ---
## # Selected controls = 8
## Kept (baseline)   : GP_population, GP_lit, RES00_gender
## Dropped (baseline): GP_sc, GP_st, GP_nbvillages, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_sc,
## Added (non-base)  : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI, CENSUS_PCA2001_tot_lit, CENSUS_VD2001_medi_
```

```r
cat("\n==== END ====\n")
```

```
##
## ==== END ====
```

Replication of Table 4 using double lasso.

```
################################ TABLE 4 ########################################

## MACROS AND DATA ##

gpcontrols15 <- c(
  "GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
  "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
  "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st", "RES15_obc", "RES15_sc", "RE
)

# Full list of candidates as controls
full_controls_candidates_table4 <- c(
  gpcontrols15
  #"CAND_vil_illit",
  #"CAND_vil_aglb",
  #"CAND_village_SCST"
  #"CAND_village_index"
)

# Dependent variables
dep_vars_table4 <- c(
  "ELEC15_nbcands", "ELEC15_incum10_running", "ELEC15_voteshare_incum10",
  "ELEC15_prop_cand2010", "ELEC15_voteshare_cand2010", "ELEC15_prop_female",
  "ELEC15_voteshare_female", "ELEC15_prop_nongen", "ELEC15_voteshare_nongen"
)

# Treatment variables
outregvar2 <- c("INT_treatment", "RES05_gender", "X_anytr_genderres05")

# Loading and filtering the data
data <- read_dta("~/work/Electoral data 2015 cleaned.dta")
data_filtered <- data %>%
  filter(RES10_gender == 0, GP_tag == 1, RES15_gender == 0) %>%
  mutate(
    INC10_can_run = 1,
    INC10_can_run = ifelse(ELEC10_won_female == 0 & RES15_gender == 1, 0, INC10_can_run),
    INC10_can_run = ifelse(ELEC10_won_sc == 0 & RES15_sc == 1, 0, INC10_can_run),
    INC10_can_run = ifelse(ELEC10_won_st == 0 & RES15_st == 1, 0, INC10_can_run),
    district = as.factor(district)
  )

# Interaction variables
for (var in c("INT_treatment", "X_anytr_genderres05", "RES05_gender")) {
  data_filtered <- data_filtered %>%
    mutate(!!paste0("X15_", var) := get(var) * (RES15_gender == 1))
}

# Checking NA among candidates
cat("NA in candidates :\n")
```

```
## NA in candidates :
```

```r
print(colSums(is.na(data_filtered[, full_controls_candidates_table4])))
```

```
## GP_population        GP_lit         GP_sc         GP_st GP_nbvillages 
##             0             0             0             0             0 
##  RES00_gender     RES00_obc      RES00_sc      RES00_st     RES10_obc 
##             0             0             0             0             0 
##      RES10_sc      RES10_st     RES05_obc      RES05_sc      RES05_st 
##             0             0             0             0             0 
##     RES15_obc      RES15_sc      RES15_st 
##             0             0             0
```

```r
# Filter NA for candidates and key variables
data_filtered <- data_filtered %>%
  filter(complete.cases(!!!syms(full_controls_candidates_table4)),
         complete.cases(!!!syms(dep_vars_table4)),
         complete.cases(!!!syms(outregvar2)))


## FUNCTIONS ##

# Function to extract names of non zero coefficients
get_nonzero_names <- function(cf, drop = "(Intercept)") {
  cf_mat <- as.matrix(cf)
  nz <- which(cf_mat != 0)
  vars <- rownames(cf_mat)[nz]
  setdiff(vars, drop)
}

# Double lasso function
double_lasso_fit <- function(data_df, y_name, d_names, x_candidates, baseline_names = NULL, fe_name = "
  # fixed effects
  if (!is.null(fe_name)) {
    fe_vec <- data_df[[fe_name]]
    fe_mat <- model.matrix(~ fe_vec - 1)
    colnames(fe_mat) <- paste0("fe_", levels(fe_vec))
  } else {
    fe_mat <- NULL
  }

  # candidate controls
  X_all <- data_df %>% dplyr::select(all_of(x_candidates))
  X_mat <- model.matrix(~ . - 1, data = X_all)

  # filter NA rows for X_mat
  keep_rows_X <- complete.cases(X_mat)

  # filter NA rows for fe_mat
  if (!is.null(fe_mat)) {
    keep_rows_FE <- complete.cases(fe_mat)
    if (length(keep_rows_X) != length(keep_rows_FE)) {
      stop("Error : X_mat and fe_mat dimensions do not match.")
    }
    keep_rows <- keep_rows_X & keep_rows_FE
  } else {
```

```
  keep_rows <- keep_rows_X
}

# filter
X_mat <- X_mat[keep_rows, , drop = FALSE]
if (!is.null(fe_mat)) {
  fe_mat <- fe_mat[keep_rows, , drop = FALSE]
}

# baseline controls
baseline_idx <- if (!is.null(baseline_names)) {
  which(colnames(X_mat) %in% baseline_names)
} else integer(0)

# add fixed effects X_mat
if (!is.null(fe_mat)) {
  X_mat <- cbind(X_mat, fe_mat)
  fe_idx <- (ncol(X_mat) - ncol(fe_mat) + 1):ncol(X_mat)
} else {
  fe_idx <- integer(0)
}

# dependent and treatment variables
missing_d <- setdiff(d_names, names(data_df))
if (length(missing_d) > 0) {
  for (nm in missing_d) {
    parts <- strsplit(nm, ":", fixed = TRUE)[[1]]
    stopifnot(all(parts %in% names(data_df)))
    data_df[[nm]] <- data_df[[parts[1]]] * data_df[[parts[2]]]
  }
}

y_vec <- as.numeric(data_df[[y_name]])
D_mat <- as.matrix(data_df %>% dplyr::select(all_of(d_names)))
if (is.null(ncol(D_mat))) D_mat <- matrix(D_mat, ncol = 1)

# filter y_vec, D_mat, and se_cluster
y_vec <- y_vec[keep_rows]
D_mat <- D_mat[keep_rows, , drop = FALSE]
if (!is.null(se_cluster)) se_cluster <- se_cluster[keep_rows]

# checks
y_constant <- length(unique(y_vec)) < 2
d_constant <- apply(D_mat, 2, function(z) length(unique(z)) < 2)

# selecting the controls Y ~ D + X
sel_y <- character(0); lambda_y <- NA
if (!y_constant) {
  X_y <- cbind(D_mat, X_mat)
  pf_y <- c(rep(0, ncol(D_mat)), rep(1, ncol(X_mat)))
  if (length(fe_idx) > 0) pf_y[ncol(D_mat) + fe_idx] <- 0

  if (all(pf_y == 0)) {
    sel_y <- character(0)
    lambda_y <- NA
  } else {
    set.seed(123)
    fit1 <- tryCatch({
      cv.glmnet(X_y, y_vec, alpha = 1, family = family, penalty.factor = pf_y)
    }, error = function(e) NULL)

    if (!is.null(fit1)) {
      coef1 <- coef(fit1, s = "lambda.min")
      cf_mat1 <- as.matrix(coef1)
      nz1 <- which(cf_mat1 != 0)
      sel_y <- rownames(cf_mat1)[nz1]
      sel_y <- setdiff(sel_y, c("(Intercept)", colnames(D_mat)))
      lambda_y <- fit1$lambda.min
    } else {
      sel_y <- character(0)
      lambda_y <- NA
    }
  }
}

# selecting controls for D_k ~ X
pf_d <- rep(1, ncol(X_mat))
if (length(baseline_idx) > 0) pf_d[baseline_idx] <- 1
if (length(fe_idx) > 0) pf_d[fe_idx] <- 0

sel_d <- character(0)
d_lambdas <- rep(NA, ncol(D_mat))

if (!all(pf_d == 0)) {
  for (k in seq_len(ncol(D_mat))) {
    if (d_constant[k]) next
    set.seed(123)
    fitk <- tryCatch({
      cv.glmnet(X_mat, D_mat[, k], alpha = 1, family = "gaussian", penalty.factor = pf_d)
    }, error = function(e) NULL)

    if (!is.null(fitk)) {
      coefk <- coef(fitk, s = "lambda.min")
      cf_matk <- as.matrix(coefk)
      nz_k <- which(cf_matk != 0)
      sel_k <- rownames(cf_matk)[nz_k]
      sel_k <- setdiff(sel_k, "(Intercept)")
      sel_d <- union(sel_d, sel_k)
      d_lambdas[k] <- fitk$lambda.min
    }
  }
}

# union of selected controls
selected_controls <- union(sel_y, sel_d)
if (length(fe_idx) > 0) selected_controls <- union(selected_controls, colnames(X_mat)[fe_idx])
```

```
  # final regression
  X_final <- X_mat[, selected_controls, drop = FALSE]
  final_df <- data.frame(Y = y_vec, D_mat, X_final)
  colnames(final_df)[2:(1 + ncol(D_mat))] <- colnames(D_mat)

  rhs <- paste(c(colnames(D_mat), colnames(X_final)), collapse = " + ")
  f_final <- as.formula(paste("Y ~", rhs))
  lm_final <- lm(f_final, data = final_df)

  # standard errors
  vc <- if (!is.null(se_cluster)) {
    sandwich::vcovCL(lm_final, cluster = se_cluster)
  } else {
    sandwich::vcovHC(lm_final, type = "HC1")
  }

  coefs_rob <- lmtest::coeftest(lm_final, vcov = vc)

  list(
    selected_controls = selected_controls,
    model = lm_final,
    robust_coefs = coefs_rob,
    lambda_y = lambda_y,
    d_lambdas = d_lambdas,
    y_constant = y_constant,
    d_constant = d_constant
  )
}

# Function for Wald tests
calculate_tests <- function(model) {
  pval1 <- tryCatch(car::linearHypothesis(model, "RES05_gender = 0")$`Pr(>F)`[2], error = function(e) N
  pval2 <- tryCatch(car::linearHypothesis(model, "X_anytr_genderres05 = 0")$`Pr(>F)`[2], error = functi
  pval3 <- tryCatch(car::linearHypothesis(model, "INT_treatment + X_anytr_genderres05 = 0")$`Pr(>F)`[2]
  return(list(pval1 = round(pval1, 2), pval2 = round(pval2, 2), pval3 = round(pval3, 2)))
}

# Lists to stock results
models_list_table4 <- list()
control_means_table4 <- numeric(length(dep_vars_table4))
test_results_table4 <- list()

## LOOP ##

# Loop on dependent variables
for (i in seq_along(dep_vars_table4)) {
  dep_var <- dep_vars_table4[i]
  cat(paste("\n=== Dependent variable :", dep_var, "===\n"))

  # control mean
  control_mean <- data_filtered %>%
    filter(INT_treatment == 0, RES05_gender == 0) %>%
    summarise(mean_val = mean(!!sym(dep_var), na.rm = TRUE)) %>%
    pull(mean_val) %>%
    round(2)
  control_means_table4[i] <- control_mean
  cat(paste("    Control mean (previously unreserved) :", control_mean, "\n"))

  # double lasso
  results <- double_lasso_fit(
    data_df = data_filtered,
    y_name = dep_var,
    d_names = outregvar2,
    x_candidates = full_controls_candidates_table4,
    baseline_names = gpcontrols15,
    fe_name = "district",
    se_cluster = data_filtered$district
  )

  models_list_table4[[dep_var]] <- results$model
  test_results_table4[[dep_var]] <- calculate_tests(results$model)
}
```

```
##
## === Dependent variable : ELEC15_nbcands ===
##     Control mean (previously unreserved) : 7.83
##
## === Dependent variable : ELEC15_incum10_running ===
##     Control mean (previously unreserved) : 0
##
## === Dependent variable : ELEC15_voteshare_incum10 ===
##     Control mean (previously unreserved) : 0
##
## === Dependent variable : ELEC15_prop_cand2010 ===
##     Control mean (previously unreserved) : 0.02
##
## === Dependent variable : ELEC15_voteshare_cand2010 ===
##     Control mean (previously unreserved) : 3.05
##
## === Dependent variable : ELEC15_prop_female ===
##     Control mean (previously unreserved) : 0.08
##
## === Dependent variable : ELEC15_voteshare_female ===
##     Control mean (previously unreserved) : 8.1
##
## === Dependent variable : ELEC15_prop_nongen ===
##     Control mean (previously unreserved) : 0.8
##
## === Dependent variable : ELEC15_voteshare_nongen ===
##     Control mean (previously unreserved) : 86.65
```

```
# Table of coefficients for treatment variables
tidy_list_table4 <- map(models_list_table4, ~ broom::tidy(.x))
tidy_list_table4 <- Map(function(df, id) { df$model_id <- id; df }, tidy_list_table4, names(models_list_
treat_table_table4 <- bind_rows(tidy_list_table4) %>%
  filter(term %in% outregvar2) %>%
  mutate(
    dep_var = model_id,
    est_se = sprintf("%.3f (%.3f)", estimate, std.error)
  ) %>%
  select(dep_var, term, est_se, p.value)
cat("\n==== TREATMENT VARIABLES COEFFICIENTS ====\n")
```

```
##
## ==== TREATMENT VARIABLES COEFFICIENTS ====
```

```
print(treat_table_table4, row.names = FALSE, n=Inf)
```

```
## # A tibble: 27 x 4
##    dep_var                   term                est_se           p.value
##    <chr>                     <chr>               <chr>              <dbl>
##  1 ELEC15_nbcands            INT_treatment       0.297 (1.130)      0.794
##  2 ELEC15_nbcands            RES05_gender        0.657 (1.075)      0.543
##  3 ELEC15_nbcands            X_anytr_genderres05 -2.243 (1.745)     0.203
##  4 ELEC15_incum10_running    INT_treatment       0.051 (0.046)      0.276
##  5 ELEC15_incum10_running    RES05_gender        0.092 (0.044)      0.0389
##  6 ELEC15_incum10_running    X_anytr_genderres05 -0.094 (0.071)     0.190
##  7 ELEC15_voteshare_incum10  INT_treatment       2.131 (1.194)      0.0782
##  8 ELEC15_voteshare_incum10  RES05_gender        1.752 (1.131)      0.125
##  9 ELEC15_voteshare_incum10  X_anytr_genderres05 -2.568 (1.822)     0.163
## 10 ELEC15_prop_cand2010      INT_treatment       0.011 (0.021)      0.607
## 11 ELEC15_prop_cand2010      RES05_gender        -0.019 (0.019)     0.329
## 12 ELEC15_prop_cand2010      X_anytr_genderres05 0.014 (0.031)      0.668
## 13 ELEC15_voteshare_cand2010 INT_treatment       0.496 (2.865)      0.863
## 14 ELEC15_voteshare_cand2010 RES05_gender        -3.038 (2.714)     0.266
## 15 ELEC15_voteshare_cand2010 X_anytr_genderres05 3.384 (4.373)      0.441
## 16 ELEC15_prop_female        INT_treatment       0.183 (0.080)      0.0254
## 17 ELEC15_prop_female        RES05_gender        0.049 (0.076)      0.522
## 18 ELEC15_prop_female        X_anytr_genderres05 -0.118 (0.122)     0.338
## 19 ELEC15_voteshare_female   INT_treatment       17.418 (8.793)     0.0512
## 20 ELEC15_voteshare_female   RES05_gender        10.074 (8.329)     0.230
## 21 ELEC15_voteshare_female   X_anytr_genderres05 -16.803 (13.421)   0.214
## 22 ELEC15_prop_nongen        INT_treatment       0.076 (0.075)      0.316
## 23 ELEC15_prop_nongen        RES05_gender        -0.078 (0.072)     0.280
## 24 ELEC15_prop_nongen        X_anytr_genderres05 0.148 (0.119)      0.219
## 25 ELEC15_voteshare_nongen   INT_treatment       6.453 (8.176)      0.433
## 26 ELEC15_voteshare_nongen   RES05_gender        -7.192 (7.643)     0.350
## 27 ELEC15_voteshare_nongen   X_anytr_genderres05 8.480 (12.677)     0.506
```

```
# Comparing selected controls vs baseline
baseline_controls <- gpcontrols15
selected_controls_list <- map(models_list_table4, ~ {
  vars <- names(coef(.x))[-1]
  setdiff(vars, outregvar2)
})
controls_diff <- tibble::tibble(
  dep_var = dep_vars_table4,
  selected = selected_controls_list
) %>%
  mutate(
    kept = map(selected, ~ intersect(.x, baseline_controls)),
    dropped = map(selected, ~ setdiff(baseline_controls, .x)),
    added = map(selected, ~ setdiff(.x, baseline_controls)),
    n_selected = map_int(selected, length)
  )
cat("\n==== COMPARING CONTROLS (baseline vs double lasso) ====\n")
```

```
##
## ==== COMPARING CONTROLS (baseline vs double lasso) ====
```

```
for (r in 1:nrow(controls_diff)) {
  cat("\n--- Dependent variable :", controls_diff$dep_var[r], "---\n")
  cat("# Selected controls =", controls_diff$n_selected[r], "\n")
  cat("Kept (baseline)   :", paste(controls_diff$kept[[r]], collapse = ", "), "\n")
  cat("Dropped (baseline)   :", paste(controls_diff$dropped[[r]], collapse = ", "), "\n")
  cat("Added (non-baseline) :", paste(controls_diff$added[[r]], collapse = ", "), "\n")
}
```

```
##
## --- Dependent variable : ELEC15_nbcands ---
## # Selected controls = 12
## Kept (baseline)   : GP_population, GP_nbvillages, RES10_sc, RES15_obc, GP_st, RES00_gender, RES05_sc
## Dropped (baseline)   : GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_st, RES05_obc,
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_incum10_running ---
## # Selected controls = 9
## Kept (baseline)   : GP_st, GP_nbvillages, RES00_gender, RES05_sc, RES05_st, RES15_st
## Dropped (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_voteshare_incum10 ---
## # Selected controls = 9
## Kept (baseline)   : GP_st, GP_nbvillages, RES00_gender, RES05_sc, RES05_st, RES15_st
## Dropped (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_prop_cand2010 ---
## # Selected controls = 9
## Kept (baseline)   : RES05_sc, GP_st, GP_nbvillages, RES00_gender, RES05_st, RES15_st
## Dropped (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_voteshare_cand2010 ---
## # Selected controls = 9
## Kept (baseline)   : RES05_sc, GP_st, GP_nbvillages, RES00_gender, RES05_st, RES15_st
## Dropped (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_prop_female ---
## # Selected controls = 9
## Kept (baseline)   : GP_st, GP_nbvillages, RES00_gender, RES05_sc, RES05_st, RES15_st
## Dropped (baseline)   : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_voteshare_female ---
## # Selected controls = 9
## Kept (baseline)    : GP_st, GP_nbvillages, RES00_gender, RES05_sc, RES05_st, RES15_st
## Dropped (baseline)    : GP_population, GP_lit, GP_sc, RES00_obc, RES00_sc, RES00_st, RES10_obc, RES10_
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_prop_nongen ---
## # Selected controls = 14
## Kept (baseline)    : GP_lit, GP_sc, RES00_obc, RES10_obc, RES15_sc, RES15_st, GP_st, GP_nbvillages, R
## Dropped (baseline)    : GP_population, RES00_sc, RES00_st, RES10_sc, RES10_st, RES05_obc, RES15_obc
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
##
## --- Dependent variable : ELEC15_voteshare_nongen ---
## # Selected controls = 19
## Kept (baseline)    : GP_lit, GP_sc, GP_st, GP_nbvillages, RES00_gender, RES00_sc, RES00_st, RES10_obc
## Dropped (baseline)    : GP_population, RES00_obc
## Added (non-baseline) : fe_BHARATPUR, fe_DHOLPUR, fe_KARAULI
```

```r
cat("\n==== END ====\n")
```

```
##
## ==== END ====
```

Replication of Table 5 using double lasso.

```r
##################################### TABLE 5 #################################

## PREPARING THE DATA ##

# Loading and merging the data
path <- file.path("~/work")
electoral <- read_dta(file.path(path, "Electoral data cleaned.dta")) %>%
  select(district, ps, gp, starts_with("GP_"), starts_with("RES"), starts_with("std_HH_NREGA"), index_e
  distinct()
household <- read_dta(file.path(path, "Household survey data cleaned.dta"))

df <- household %>%
  left_join(electoral, by = c("district", "ps", "gp")) %>%
  filter(!is.na(index_empl_pre_svysample))
```

```
## Warning in left_join(., electoral, by = c("district", "ps", "gp")): Detected an unexpected many-to-ma
## i Row 1 of `x` matches multiple rows in `y`.
## i Row 1 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```r
# Deleting the duplicates after the join
dups <- names(df) %>% str_subset("\\.x$")
for (vx in dups) {
  base <- str_remove(vx, "\\.x$")
  vy <- paste0(base, ".y")
  df <- df %>%
    mutate(!!base := coalesce(
      if (vx %in% names(df)) .data[[vx]] else NULL,
      if (vy %in% names(df)) .data[[vy]] else NULL
    ))
}
```

```
df <- df %>% select(-matches("\\.(x|y)$"))

# Transformation of the variables at individual level
long_vars <- c("A_age", "A_educ", "A_literacy", "D_NREGA_work",
               "E_know_minimumwage", "E_know_maximumdays", "E_know_sarpanchrole_projects",
               "E_know_sarpanchrole_jobcard", "E_know_sarpanchrole_work", "E_know_jobcardapplication",
               "E_know_waitingdays", "E_know_unemploymentallowance", "E_know_postofficepay",
               "E_rate_NREGAimplementation", "E_rate_NREGAimplementation_g", "E_rate_NREGAimplementatio
               "E_rate_sarpanchperformance", "E_rate_sarpanchperformance_g", "E_rate_sarpanchperformanc
               "F_rank_publicgoods_road", "F_rank_publicgoods_pump", "F_rank_publicgoods_school",
               "F_rate_publicgoods_road", "F_rate_publicgoods_pump", "F_rate_publicgoods_school",
               "F_optimistic_sarpanch", "F_optimistic_govprograms")

base_dups <- long_vars[long_vars %in% names(df)]
df <- df %>% select(-all_of(base_dups))

# Dealing with NA
df_long <- df %>%
  mutate(TEMP_id = row_number()) %>%
  pivot_longer(
    cols = matches(paste0("^(", paste(long_vars, collapse = "|"), ")_(m|f)$")),
    names_to = c(".value", "gender"),
    names_pattern = "^(.*)_(m|f)$"
  ) %>%
  mutate(
    # just create an indicator (not making it become a 0 when NA)
    C_I_AgeMissing = is.na(A_age),
    A_age = ifelse(is.na(A_age), median(A_age, na.rm = TRUE), A_age)
  )

# Index variables
df_long <- df_long %>%
  mutate(
    C_I_AgeBelow25 = A_age < 25,
    C_I_Age2535 = between(A_age, 25, 34),
    C_I_Age3545 = between(A_age, 35, 44),
    C_I_AgeAbove45 = A_age >= 45,
    C_I_Female = gender == "f",
    C_I_Literate = A_literacy == 4,
    C_I_EducNone = A_educ == 0,
    C_I_EducPrimary = A_educ > 0 & A_educ <= 5,
    C_I_EducLowerSec = A_educ > 5 & A_educ <= 9,
    C_I_EducUpperSec = A_educ > 9 & A_educ <= 12,
    C_I_EducTertiary = A_educ > 12 & !is.na(A_educ),
    C_I_Missing = if_any(c(A_educ, A_literacy), is.na),
    C_H_bpl = H_bpl == 1,
    C_H_ownland = H_ownland == 1,
    C_H_hindu = H_religion == 1,
    C_H_CasteGen = H_caste %in% c(1, 5),
    C_H_CasteOBC = H_caste %in% c(2, 6),
    C_H_CasteSC = H_caste == 3,
    C_H_CasteST = H_caste == 4,
    C_H_Missing = if_any(c(H_bpl, H_ownland, H_religion, H_caste), is.na)
```

```r
  )


# Converting dummies to numerals
indcontrols <- grep("^C_I_", names(df_long), value = TRUE)
hhcontrols <- grep("^C_H_", names(df_long), value = TRUE)
dummy_vars <- c(indcontrols, hhcontrols)
df_long <- df_long %>%
  mutate(across(all_of(dummy_vars), ~as.numeric(replace_na(., 0))))

# Variables' normalisation
to_z <- grep("^(E_know_|F_rate_|E_rate)", names(df_long), value = TRUE)
ref_mu <- df_long %>%
  filter(INT_treatment == 0, RES05_gender == 0) %>%
  summarise(across(all_of(to_z), mean, na.rm = TRUE))
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(all_of(to_z), mean, na.rm = TRUE)`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```r
ref_sd <- df_long %>%
  filter(INT_treatment == 0, RES05_gender == 0) %>%
  summarise(across(all_of(to_z), sd, na.rm = TRUE))
df_long[to_z] <- map2_dfc(df_long[to_z], names(df_long[to_z]), \(x, v) (x - ref_mu[[v]]) / ref_sd[[v]])

# Composite variables
df_long <- df_long %>%
  mutate(
    E_know_nregarules = rowMeans(cbind(E_know_minimumwage, E_know_maximumdays), na.rm = TRUE),
    E_know_sarpanchrole = rowMeans(pick(starts_with("E_know_sarpanchrole_")), na.rm = TRUE),
    E_rate_nrega = rowMeans(cbind(E_rate_NREGAimplementation, E_rate_sarpanchperformance), na.rm = TRUE
    F_rate_publicgoods = rowMeans(pick(starts_with("F_rate_publicgoods_")), na.rm = TRUE),
    TEMP_index = index_empl_pre_svysample,
    TEMP_X_res_index = RES05_gender * TEMP_index,
    TEMP_X_anytr_index = INT_treatment * TEMP_index,
    TEMP_X_anytr_res_index = INT_treatment * RES05_gender * TEMP_index
  )

# Complete list of candidates
gpcontrols <- c("GP_population", "GP_lit", "GP_sc", "GP_st", "GP_nbvillages",
                "RES00_gender", "RES00_obc", "RES00_sc", "RES00_st",
                "RES10_obc", "RES10_sc", "RES10_st", "RES05_obc", "RES05_sc", "RES05_st")
#census_controls <- c("CENSUS_PCA2001_tot_pop", "CENSUS_PCA2001_tot_lit", "CENSUS_PCA2001_tot_sc",
#"CENSUS_PCA2001_tot_st", "CENSUS_PCA2001_tot_aglb", "CENSUS_PCA2001_tot_nm_hh",
#"CENSUS_PCA2001_tot_cult", "CENSUS_VD2001_power_dom", "CENSUS_VD2001_drnk_wat_f",
#"CENSUS_VD2001_edu_fac", "CENSUS_VD2001_medi_fac")
```

```r
full_controls_candidates <- c(gpcontrols,
                              #census_controls,
                              indcontrols, hhcontrols)

# Dependent variables
dep_set1 <- c("E_know_nregarules", "E_know_sarpanchrole", "E_rate_nrega", "F_rate_publicgoods")
dep_set2 <- c("index_empl_pre_svysample", dep_set1)

# Deleting NA
df_long_clean <- df_long %>%
  filter(!is.na(INT_treatment), !is.na(TEMP_index), !is.na(TEMP_X_anytr_index)) %>%
  filter_at(vars(dep_set2), all_vars(!is.na(.)))
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(dep_set2)
##
##   # Now:
##   data %>% select(all_of(dep_set2))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
# Checking amount of observations
table(df_long_clean$RES05_gender)
```

```
##
##     0     1
## 12538  6724
```

```r
## DOUBLE LASSO ##

double_lasso_fit_forced <- function(data_df, y_name, d_names, x_candidates, baseline_names = NULL) {
  # checks
  if (!y_name %in% names(data_df)) stop(paste("Missing dependent variable :", y_name))
  missing_d <- setdiff(d_names, names(data_df))
  if (length(missing_d) > 0) stop(paste("Missing variables :", paste(missing_d, collapse = ", ")))
  x_candidates <- x_candidates[x_candidates %in% names(data_df)]
  if (length(x_candidates) == 0) stop("No valid candidate.")

  # filter NA
  keep_rows <- complete.cases(data_df[, c(y_name, d_names, x_candidates)])
  if (sum(keep_rows) < 10) {
    warning(paste("Not enough obs after filter for", y_name, ". Remaining obs:", sum(keep_rows)))
    return(NULL)
  }

  # check if constant
  if (sd(data_df[[y_name]][keep_rows], na.rm = TRUE) < 0.01) {
    warning(paste("Constant or quasi-constant dependent variable", y_name))
    return(NULL)
  }
```

```r
# X and D matrices
X_all <- model.matrix(~ . - 1, data = data_df[keep_rows, x_candidates, drop = FALSE])
D_mat <- model.matrix(~ . - 1, data = data_df[keep_rows, d_names, drop = FALSE])
y_vec <- data_df[[y_name]][keep_rows]

# D_mat not constant
if (any(apply(D_mat, 2, function(x) sd(x, na.rm = TRUE) < 0.01))) {
  warning(paste("Insterest variables constant/quasi-constant for ", y_name))
  return(NULL)
}

# Double lasso for Y ~ D + X
X_y <- cbind(D_mat, X_all)
pf_y <- c(rep(0, ncol(D_mat)), rep(1, ncol(X_all)))

set.seed(123)
fit1 <- tryCatch({
  cv.glmnet(X_y, y_vec, alpha = 1, penalty.factor = pf_y)
}, error = function(e) {
  warning(paste("Error in cv.glmnet pour Y ~ D + X :", e$message))
  NULL
})

sel_y <- character(0)
if (!is.null(fit1)) {
  coef1 <- coef(fit1, s = "lambda.min")
  sel_y <- setdiff(rownames(coef1)[which(coef1 != 0)], "(Intercept)")
} else {
  warning("Fail on first double lasso stage Y ~ D + X")
  return(NULL)
}

# Double lasso for D_k ~ X
sel_d <- character(0)
for (k in 1:ncol(D_mat)) {
  pf_d <- rep(1, ncol(X_all))

  fitk <- tryCatch({
    cv.glmnet(X_all, D_mat[, k], alpha = 1, penalty.factor = pf_d)
  }, error = function(e) {
    warning(paste("Error in cv.glmnet pour D_k ~ X :", e$message))
    NULL
  })

  if (!is.null(fitk)) {
    coefk <- coef(fitk, s = "lambda.min")
    sel_k <- setdiff(rownames(coefk)[which(coefk != 0)], "(Intercept)")
    sel_d <- union(sel_d, sel_k)
  } else {
    warning(paste("Fail of the lasso for D_k ~ X, variable :", colnames(D_mat)[k]))
    return(NULL)
  }
}
```

```
  # Final selection of controls
  selected_controls <- union(sel_y, sel_d)
  if (length(selected_controls) == 0) selected_controls <- colnames(X_all)

  # Final model: force inclusion of interest variables
  final_data <- data.frame(Y = y_vec, D_mat, X_all[, selected_controls, drop = FALSE])

  # Build formula with interest variables and selected controls
  rhs_terms <- c(colnames(D_mat), colnames(X_all[, selected_controls, drop = FALSE]))

  # Add interactions
  if ("INT_treatment" %in% colnames(D_mat) && "TEMP_index" %in% colnames(D_mat)) {
    rhs_terms <- c(rhs_terms, "INT_treatment:TEMP_index")
  }
  if ("INT_treatment" %in% colnames(D_mat) && "RES05_gender" %in% colnames(D_mat)) {
    rhs_terms <- c(rhs_terms, "INT_treatment:RES05_gender")
  }
  if ("TEMP_index" %in% colnames(D_mat) && "RES05_gender" %in% colnames(D_mat)) {
    rhs_terms <- c(rhs_terms, "TEMP_index:RES05_gender")
  }
  if ("INT_treatment" %in% colnames(D_mat) && "TEMP_index" %in% colnames(D_mat) && "RES05_gender" %in%
    rhs_terms <- c(rhs_terms, "INT_treatment:TEMP_index:RES05_gender")
  }
  rhs_terms <- unique(rhs_terms)

  if (length(rhs_terms) == 0) {
    warning("No valid variable.")
    return(NULL)
  }

  f_final <- as.formula(paste("Y ~", paste(rhs_terms, collapse = " + ")))

  # Model estimation
  model <- tryCatch({
    lm(f_final, data = final_data)
  }, error = function(e) {
    warning(paste("Error in final model estimation:", e$message))
    NULL
  })

  if (is.null(model)) return(NULL)

  list(
    model = model,
    selected_controls = selected_controls,
    coefficients = coef(model),
    se = sqrt(diag(vcovHC(model, type = "HC1")))
  )
}



## EXECUTION OF REGRESSIONS ##
```

```r
# Initialisation
models_list_forced <- vector("list", length(dep_set2) * 2)
selected_controls_list_forced <- vector("list", length(dep_set2) * 2)
results_list_forced <- vector("list", length(dep_set2) * 2)

# Estimation loop
for (x in 0:1) {  # RES05_gender
  for (i in seq_along(dep_set2)) {
    y_name <- dep_set2[i]
    index <- x * length(dep_set2) + i

    cat("\n--- Estimation for", y_name, "| RES05_gender =", x, "---\n")

    filtered_data <- df_long_clean %>% filter(RES05_gender == x)
    if (nrow(filtered_data) < 10) {
      warning(paste("Not enough data for RES05_gender =", x, "et", y_name))
      next
    }

    # Interest variables
    d_names <- c("INT_treatment", "TEMP_index", "RES05_gender")

    # Estimation
    result <- tryCatch({
      double_lasso_fit_forced(
        data_df = filtered_data,
        y_name = y_name,
        d_names = d_names,
        x_candidates = full_controls_candidates
      )
    }, error = function(e) {
      warning(paste("Error :", e$message))
      NULL
    })

    if (!is.null(result) && !is.null(result$model)) {
      models_list_forced[[index + 1]] <- result$model
      selected_controls_list_forced[[index + 1]] <- result$selected_controls
      results_list_forced[[index + 1]] <- result
    } else {
      warning(paste("Model not estimated", y_name, "| RES05_gender =", x))
      models_list_forced[[index + 1]] <- NULL
      selected_controls_list_forced[[index + 1]] <- character(0)
      results_list_forced[[index + 1]] <- NULL
    }
  }
}
```

```
##
## --- Estimation for index_empl_pre_svysample | RES05_gender = 0 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for index_empl_pre_svysample
## Warning: Model not estimated index_empl_pre_svysample | RES05_gender = 0

##
## --- Estimation for E_know_nregarules | RES05_gender = 0 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_know_nregarules

## Warning: Model not estimated E_know_nregarules | RES05_gender = 0

##
## --- Estimation for E_know_sarpanchrole | RES05_gender = 0 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_know_sarpanchrole

## Warning: Model not estimated E_know_sarpanchrole | RES05_gender = 0

##
## --- Estimation for E_rate_nrega | RES05_gender = 0 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_rate_nrega

## Warning: Model not estimated E_rate_nrega | RES05_gender = 0

##
## --- Estimation for F_rate_publicgoods | RES05_gender = 0 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for F_rate_publicgoods

## Warning: Model not estimated F_rate_publicgoods | RES05_gender = 0

##
## --- Estimation for index_empl_pre_svysample | RES05_gender = 1 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for index_empl_pre_svysample

## Warning: Model not estimated index_empl_pre_svysample | RES05_gender = 1

##
## --- Estimation for E_know_nregarules | RES05_gender = 1 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_know_nregarules

## Warning: Model not estimated E_know_nregarules | RES05_gender = 1

##
## --- Estimation for E_know_sarpanchrole | RES05_gender = 1 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_know_sarpanchrole

## Warning: Model not estimated E_know_sarpanchrole | RES05_gender = 1

##
## --- Estimation for E_rate_nrega | RES05_gender = 1 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for E_rate_nrega

## Warning: Model not estimated E_rate_nrega | RES05_gender = 1
```

```
##
## --- Estimation for F_rate_publicgoods | RES05_gender = 1 ---

## Warning in double_lasso_fit_forced(data_df = filtered_data, y_name = y_name, :
## Insterest variables constant/quasi-constant for F_rate_publicgoods

## Warning: Model not estimated F_rate_publicgoods | RES05_gender = 1
```

```
# Nombre de modèles estimés avec succès
valid_models <- sapply(models_list_forced, function(x) !is.null(x))
cat("\nSuccessful estimations :", sum(valid_models), "\n")
```

```
##
## Successful estimations : 0
```

```
# Affichage des résultats
if (sum(valid_models) > 0) {
  cat("\n==== MODELS ====\n")
  for (i in which(valid_models)) {
    cat("Model", i, ":", names(dep_set2)[((i-1) %% length(dep_set2)) + 1],
        "| RES05_gender =", (i-1) %/% length(dep_set2), "\n")
    print(summary(models_list_forced[[i]]))
  }
} else {
  cat("\nNo model estimated.\n")
}
```

```
##
## No model estimated.
```

```
# Affichage des contrôles sélectionnés
cat("\n==== CONTROLS SELECTED BY DOUBLE LASSO ====\n")
```

```
##
## ==== CONTROLS SELECTED BY DOUBLE LASSO ====
```

```
for (i in seq_along(selected_controls_list_forced)) {
  if (length(selected_controls_list_forced[[i]]) > 0) {
    cat("Model", i, ":", names(dep_set2)[((i-1) %% length(dep_set2)) + 1],
        "| RES05_gender =", (i-1) %/% length(dep_set2), "\n")
    cat("Selected controls :", paste(selected_controls_list_forced[[i]], collapse = ", "), "\n\n")
  } else {
    cat("Model", i, ":", names(dep_set2)[((i-1) %% length(dep_set2)) + 1],
        "| RES05_gender =", (i-1) %/% length(dep_set2), "\n")
    cat("No selected controls.\n\n")
  }
}
```

```
## Model 1 : | RES05_gender = 0
## No selected controls.
##
## Model 2 : | RES05_gender = 0
## No selected controls.
##
## Model 3 : | RES05_gender = 0
## No selected controls.
##
## Model 4 : | RES05_gender = 0
## No selected controls.
##
## Model 5 : | RES05_gender = 0
## No selected controls.
##
## Model 6 : | RES05_gender = 1
## No selected controls.
##
## Model 7 : | RES05_gender = 1
## No selected controls.
##
## Model 8 : | RES05_gender = 1
## No selected controls.
##
## Model 9 : | RES05_gender = 1
## No selected controls.
##
## Model 10 : | RES05_gender = 1
## No selected controls.
##
## Model 11 : | RES05_gender = 2
## No selected controls.
```

```r
# Sélection des coefficients à afficher
coefs_to_extract <- c(
  "TEMP_index",
  "INT_treatment",
  "RES05_gender",
  "TEMP_X_anytr_index",
  "TEMP_X_res_index",
  "TEMP_X_anytr_res_index"
)

extract_and_display_coefficients <- function(models_list, selected_controls_list, dep_set2) {
  for (i in seq_along(models_list)) {
    if (!is.null(models_list[[i]])) {
      model <- models_list[[i]]
      tidy_model <- tidy(model, conf.int = TRUE)
      model_name <- names(dep_set2)[((i-1) %% length(dep_set2)) + 1]
      gender <- (i-1) %/% length(dep_set2)

      cat("\n==== Modèle", i, ":", model_name, "| RES05_gender =", gender, "====\n")

      relevant_coefs <- tidy_model %>%
        filter(term %in% coefs_to_extract) %>%
        select(term, estimate, std.error, p.value)

      if (nrow(relevant_coefs) > 0) {
        print(relevant_coefs)
      } else {
        cat("No coefficient found in the model.\n")
      }

      # Affichage des contrôles sélectionnés
      cat("\nSelected controls :\n")
      cat(paste(selected_controls_list[[i]], collapse = ", "), "\n")
    }
  }
}

extract_and_display_coefficients(models_list_forced, selected_controls_list_forced, dep_set2)
```