ECMA 31360, PSet 2: Solutions

Winston Su, Cagan Gurdal, and Bailey Meche, University of Chicago

01-23-2026

Carry-over from PSet2

([ ] out of 50p) PART I: Test Balance in the Observed Predetermined Variables (OPVs)

([ ] out of 15p) Q1: Read and Understand the NSW Application's Companion Document

Done!

This helper document demonstrates how to test an implication of URA in the NSW RCT: balance of OPVs between treated and control groups. Formally, URA implies independence between x_i and D_i, which can be written either as equality of OPV distributions

$$f_{x|D=1}(a) = f_{x|D=0}(a) \;\; \forall a \in \mathcal{X},$$

or as constant treatment assignment probabilities across OPV values

$$\Pr(D_i = 1 \mid x_i = a) = \Pr(D_i = 1 \mid x_i = a') \;\; \forall a, a' \in \mathcal{X}.$$

Because directly testing equality of multivariate distributions is demanding, the document recommends a more practical modest target: equality of means for each OPV,

$$H_0: \; \mathbb{E}[x_{j,i} \mid D_i = 1] = \mathbb{E}[x_{j,i} \mid D_i = 0] \;\; \forall j = 1, \ldots, J,$$

and explains why naïve one-at-a-time testing of these J hypotheses can over-reject due to the multiple testing problem (FWER can be much larger than α when many hypotheses are tested).

To address multiple testing, the document provides a roadmap with two main strategies:

1. Joint hypothesis testing:
   - Procedure 4 (SUR approach): write J regressions $x_{j,i} = \pi_{0,j} + \pi_{1,j} D_i + \varepsilon_{j,i}$ where $\pi_{1,j} = \mathbb{E}[x_{j,i} \mid D_i = 1] - \mathbb{E}[x_{j,i} \mid D_i = 0]$, stack them as a SUR system, estimate by (F)GLS, and test the joint null $H_0 : \pi_{1,1} = \cdots = \pi_{1,J} = 0$ using an $F_{J,Jn-K}$ statistic or equivalently the Wald statistic $S = J \cdot F \sim \chi^2_J$ (Section 5.1.1).
   - Procedure 5 (Hotelling's T^2): test the same joint equal-means hypothesis using

     $$T^2 = \frac{n_1 n_0}{n_1 + n_0} (\bar{x}_1 - \bar{x}_0)^\top C_p^{-1} (\bar{x}_1 - \bar{x}_0),$$

     and the document emphasizes the key equivalence $T^2 = S$ (Claim 8)
2. Testing that OPVs do not predict assignment (Procedure 6): regress D_i on a constant and x_i (often via an LPM) and use the overall significance F-test based on R^2,

   $$F = \frac{R^2/M}{(1 - R^2)/(n - (M + 1))},$$

   where M is the number of OPV slope parameters (Section 5.2). The document notes that in theory testing equal means and testing "OPVs don't predict assignment" are equivalent implications of URA, but in practice they may differ because both procedures rely on additional parametric/scaling choices (Section 5.3).

Finally, the document explains FWER control via Bonferroni (Procedure 7) and Holm–Bonferroni (Procedure 8) adjustments, defines FWER formally, and shows how to adjust one-at-a-time p-values so that FWER $\leq \alpha$.

([ ] out of 23p) Q2: Implement Procedure 4 (SUR Estimation followed by Joint Testing)

([ ] out of 10p) Q2.a: SUR Estimation

```
library(systemfit) # Load data and create treatment indicator
```

```
## Loading required package: Matrix

## Loading required package: car

## Loading required package: carData

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Please cite the 'systemfit' package as:
## Arne Henningsen and Jeff D. Hamann (2007). systemfit: A Package for Estimating Systems of Simultaneous Equati
##
## If you have questions, suggestions, or comments regarding the 'systemfit' package, please use a forum or 'tra
## https://r-forge.r-project.org/projects/systemfit/
```

```
treated <- read.csv("nswre74_treated.csv")
control <- read.csv("nswre74_control.csv")
treated$treat <- 1
control$treat <- 0
df <- rbind(treated, control)
opvs <- c("age","edu","nodegree","black","hisp", "married","u74","u75","re74","re75")
sur_system <- setNames( lapply(opvs, function(v) as.formula(paste(v, "~ treat"))), opvs )

#   pi0_j = control mean of OPV j
#   pi1_j = treated-control difference in means for OPV j
#   SE(pi1_j) from SUR vcov


sur_fit <- systemfit(sur_system, data = df, method = "SUR")

b_hat <- coef(sur_fit)
V_hat <- vcov(sur_fit)
nm <- names(b_hat)

# get intercept, treat entries for each equation j
```

```r
get_idx <- function(v) {
  i0 <- which(nm == paste0(v, "_(Intercept)"))
  i1 <- which(nm == paste0(v, "_treat"))
  c(i0 = i0, i1 = i1)
}

tab_q2a <- do.call(rbind, lapply(opvs, function(v) {
  idx <- get_idx(v)
  pi0 <- b_hat[idx["i0"]]
  pi1 <- b_hat[idx["i1"]]
  se1 <- sqrt(V_hat[idx["i1"], idx["i1"]])
  z   <- pi1 / se1
  p   <- 2 * pnorm(-abs(z))   # normal approx
  data.frame(
    OPV = v,
    pi0_control_mean = as.numeric(pi0),
    pi1_diff_means   = as.numeric(pi1),
    se_pi1           = as.numeric(se1),
    z_stat           = as.numeric(z),
    p_value_normal   = as.numeric(p)
  )
}))

tab_q2a
```

```
##         OPV pi0_control_mean pi1_diff_means       se_pi1      z_stat
## 1       age       25.0538462     0.76237006   0.68275109  1.11661493
## 2       edu       10.0884615     0.25748441   0.17213532  1.49582552
## 3  nodegree        0.8346154    -0.12650728   0.03934520 -3.21531660
## 4     black        0.8269231     0.01632017   0.03588617  0.45477598
## 5      hisp        0.1076923    -0.04823285   0.02716320 -1.77566909
## 6   married        0.1538462     0.03534304   0.03604844  0.98043187
## 7       u74        0.7500000    -0.04189189   0.04262210 -0.98286779
## 8       u75        0.6846154    -0.08461538   0.04582176 -1.84662032
## 9      re74     2107.0266512   -11.45295788 516.47797093 -0.02217511
## 10     re75     1266.9090145   265.14629853 303.15549725  0.87462144
##    p_value_normal
## 1     0.264159006
## 2     0.134699129
## 3     0.001303007
## 4     0.649270410
## 5     0.075787474
## 6     0.326872984
## 7     0.325672514
## 8     0.064802187
## 9     0.982308269
## 10    0.381779916
```

Script and Output

We estimate a 10-equation SUR system where each OPV is regressed on the treatment indicator. The intercept in each equation equals the control-group mean of that OPV, and the coefficient on treat equals the treated–control difference in means. The estimated residual covariance/correlation matrices show substantial cross-equation dependence for economically related OPVs (`u74`-`u75`, `re74`-`re75`, `edu`-`nodegree`), which motivates using SUR and is required for the joint balance test in subsequent questions.

([ ] out of 1p) Q2.b: Compare FGLS to OLS equation-by-equation

```r
# Q2.b: OLS equation-by-equation (for comparison)
ols_fits <- lapply(opvs, function(v) lm(as.formula(paste0(v, " ~ treat")), data = df))
names(ols_fits) <- opvs

# Compare treat coefficients
sapply(ols_fits, function(m) coef(m)["treat"])
```

```
##      age.treat      edu.treat nodegree.treat    black.treat     hisp.treat
##     0.76237006     0.25748441    -0.12650728     0.01632017    -0.04823285
##  married.treat      u74.treat      u75.treat     re74.treat     re75.treat
##     0.03534304    -0.04189189    -0.08461538   -11.45295788   265.14629853
```

```r
coef(sur_fit)[grep("treat", names(coef(sur_fit)))]
```

```
##      age_treat      edu_treat nodegree_treat    black_treat     hisp_treat
##     0.76237006     0.25748441    -0.12650728     0.01632017    -0.04823285
##  married_treat      u74_treat      u75_treat     re74_treat     re75_treat
##     0.03534304    -0.04189189    -0.08461538   -11.45295788   265.14629853
```

“Because each equation uses the same regressor matrix $(1, D_i)$, SUR and OLS deliver identical $\hat{\pi}_{1,j}$ with no efficiency gain equation-by-equation. SUR’s value here is the estimated cross-equation covariance needed to form the joint Wald/F test.

([ ] out of 3p) Q2.c: Take a closer look at the variance-covariance matrix

Script and Output

```r
V_hat <- vcov(sur_fit)
nm <- names(coef(sur_fit))
nm[1:4]  # should look like: age_(Intercept), age_treat, edu_(Intercept), edu_treat
```

```
## [1] "age_(Intercept)" "age_treat"       "edu_(Intercept)" "edu_treat"
```

```r
i01 <- 1  # pi0,1
i11 <- 2  # pi1,1
i02 <- 3  # pi0,2
i12 <- 4  # pi1,2

# (i) should be ~ 0
V_hat[i01, i11] + V_hat[i01, i01]
```

```
## [1] 1.387779e-16
```

```r
# (ii) should be ~ 0
V_hat[i01, i12] + V_hat[i01, i02]
```

```
## [1] 1.832302e-17
```

```r
# (iii) should be ~ 0
V_hat[i01, i12] - V_hat[i11, i02]
```

```
## [1] 5.334275e-17
```

We estimate, for each OPV $j = 1, \ldots, 10$,

$$X_{ij} = \pi_{0,j} + \pi_{1,j} D_i + u_{ij}, \qquad D_i \in 0, 1.$$

With regressors $(1, D_i)$, the OLS (hence the equation-by-equation component of SUR) admits the exact sample-mean form:

$$\widehat{\pi}_{0,j} = \bar{X}_{j,0}, \qquad \widehat{\pi}_{1,j} = \bar{X}_{j,1} - \bar{X}_{j,0},$$

where $\bar{X}_{j,0}$ and $\bar{X}_{j,1}$ are the sample means of X_{ij} in the control and treated subsamples, respectively.

Crucially, $\bar{X}_{j,0}$ and $\bar{X}_{k,1}$ are computed from disjoint groups of observations (controls vs treated). Under the standard framework of i.i.d. sampling and independent assignment, functions of the treated subsample are independent of functions of the control subsample, implying $Cov\left(\bar{X}_{j,0}, \bar{X}_{k,1}\right) = 0$ exact under independence; approximately 0 asymptotically under standard randomization arguments.

To show why $\widehat{\text{Cov}}[\widehat{\pi}_{0,1}, \widehat{\pi}_{1,1}] = -\widehat{\text{Var}}[\widehat{\pi}_{0,1}]$, we have $\widehat{\pi}_{0,1} = \bar{X}_{1,0}) and (\widehat{\pi}_{1,1} = \bar{X}_{1,1} - \bar{X}_{1,0}$, so

$$\text{Cov}(\widehat{\pi}_{0,1}, \widehat{\pi}_{1,1}) = \text{Cov}(\bar{X}_{1,0}, \bar{X}_{1,1}) - \text{Var}(\bar{X}_{1,0}) = 0 - \text{Var}(\bar{X}_{1,0}),$$

then $\text{Cov}(\widehat{\pi}_{0,1}, \widehat{\pi}_{1,1}) = -\text{Var}(\widehat{\pi}_{0,1})$.

To show why $\widehat{\text{Cov}}[\widehat{\pi}_{0,1}, \widehat{\pi}_{1,2}] = -\widehat{\text{Cov}}[\widehat{\pi}_{0,1}, \widehat{\pi}_{0,2}]$, we have $\widehat{\pi}_{0,1} = \bar{X}_{1,0}$ and $\widehat{\pi}_{1,2} = \bar{X}_{2,1} - \bar{X}_{2,0}$,

$$\text{Cov}(\widehat{\pi}_{0,1}, \widehat{\pi}_{1,2}) = \text{Cov}(\bar{X}_{1,0}, \bar{X}_{2,1}) - \text{Cov}(\bar{X}_{1,0}, \bar{X}_{2,0}) = 0 - \text{Cov}(\widehat{\pi}_{0,1}, \widehat{\pi}_{0,2}).$$

To show why $\widehat{\text{Cov}}[\widehat{\pi}_{0,1}, \widehat{\pi}_{1,2}] = \widehat{\text{Cov}}[\widehat{\pi}_{1,1}, \widehat{\pi}_{0,2}])$, we compute

$$\text{Cov}(\widehat{\pi} * 1, 1, \widehat{\pi} * 0, 2) = \text{Cov}(\bar{X}_{1,1} - \bar{X}_{1,0}, \bar{X}_{2,0}) = \text{Cov}(\bar{X}_{1,1}, \bar{X}_{2,0}) - \text{Cov}(\bar{X}_{1,0}, \bar{X}_{2,0}) = 0 - \text{Cov}(\bar{X}_{1,0}, \bar{X}_{2,0}),$$

which equals the expression in (ii), hence the claimed equality.

Commentary

Because each equation includes only an intercept and a binary treatment indicator, $\widehat{\pi}_{0,j}$ equals the control-group mean and $\widehat{\pi}_{1,j}$ equals the treated–control difference in means. Under unconditional random assignment, treated and control subsamples are (approximately) independent. These two facts jointly imply the sign and symmetry restrictions in the top-left 4×4 block of $\widehat{\text{Var}}(\widehat{\pi})$, and the implied equalities are verified numerically up to machine precision.

([ ] out of 4p) Q2.d: Implement Joint Test Manually

```r
# coef and vcov from SUR
b_hat <- coef(sur_fit)
V_hat <- vcov(sur_fit)
nm <- names(b_hat)

# 1) pick the J=10 treat coefficients
treat_idx <- grep("_treat$", nm)
J <- length(treat_idx)        # should be 10
K <- length(b_hat)            # should be 20

# 2) build R matrix selecting treat coefficients: R b = (pi_1,1,...,pi_1,J)'
R <- matrix(0, nrow = J, ncol = K)
for (j in 1:J) R[j, treat_idx[j]] <- 1
r0 <- rep(0, J)

# 3) compute quadratic form Q = (R b - r)' (R V R')^{-1} (R b - r)
d <- as.vector(R %*% b_hat - r0)
Q <- as.numeric(t(d) %*% solve(R %*% V_hat %*% t(R)) %*% d)
```

```
# 4) F statistic and p-value using F_{J, Jn-K}
F_stat <- Q / J
df1 <- J
n <- 445
df2 <- J*n - K                         # should be 4430
p_F <- 1 - pf(F_stat, df1 = df1, df2 = df2)

# 5) S (Wald) statistic and p-value using Chi-square_J
S_stat <- J * F_stat   # equals Q
p_S <- 1 - pchisq(S_stat, df = J)

c(J=J, K=K, n=n, df2=df2,
  F_stat=F_stat, p_value_F=p_F,
  S_stat=S_stat, p_value_S=p_S)
```

```
##            J            K            n          df2       F_stat    p_value_F
## 1.000000e+01 2.000000e+01 4.450000e+02 4.430000e+03 2.046563e+00 2.538070e-02
##       S_stat    p_value_S
## 2.046563e+01 2.514383e-02
```

Script and Output

We test the joint null hypothesis that the coefficients on `treat` are zero in all $J = 10$ equations:

$$H_0: \ \pi_{1,1} = \pi_{1,2} = \cdots = \pi_{1,10} = 0.$$

Using the companion document's expressions for Procedure 4, we compute the F statistic and the Wald (S) statistic. With $n = 445$ observations per equation and $K = 2J = 20$ parameters in the stacked system, the denominator degrees of freedom are

$$Jn - K = 10 \times 445 - 20 = 4430.$$

Our manual calculations yield

$$F = 2.0466 \quad \text{with} \quad p\text{-value} = 0.0254 \ \ \text{using } F_{10,4430},$$

and equivalently

$$S = J \cdot F = 20.4656 \quad \text{with} \quad p\text{-value} = 0.0251 \ \ \text{using } \chi^2_{10}.$$

At the 5% level, we reject H_0, indicating that the OPVs are not jointly balanced across treatment and control.

([ ] out of 4p) Q2.e: Implement Joint Test Automatically

Script and Output

```
library(car)

restr <- c(
  "age_treat = 0",
  "edu_treat = 0",
  "nodegree_treat = 0",
  "black_treat = 0",
  "hisp_treat = 0",
  "married_treat = 0",
  "u74_treat = 0",
  "u75_treat = 0",
  "re74_treat = 0",
  "re75_treat = 0"
)

# F-form
lh_F <- linearHypothesis(sur_fit, restr, test = "F")
lh_F
```

```
## Linear hypothesis test (F statistic of a Wald test)
##
## Hypothesis:
## age_treat = 0
## edu_treat = 0
## nodegree_treat = 0
## black_treat = 0
## hisp_treat = 0
## married_treat = 0
## u74_treat = 0
## u75_treat = 0
## re74_treat = 0
## re75_treat = 0
##
## Model 1: restricted model
## Model 2: sur_fit
##
##   Res.Df Df      F  Pr(>F)
## 1   4440
## 2   4430 10 2.0466 0.02538 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Wald Chi-square form
lh_S <- linearHypothesis(sur_fit, restr, test = "Chisq")
lh_S
```

```
## Linear hypothesis test (Chi^2 statistic of a Wald test)
##
## Hypothesis:
## age_treat = 0
## edu_treat = 0
## nodegree_treat = 0
## black_treat = 0
## hisp_treat = 0
## married_treat = 0
## u74_treat = 0
## u75_treat = 0
## re74_treat = 0
## re75_treat = 0
##
## Model 1: restricted model
## Model 2: sur_fit
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1   4440
## 2   4430 10 20.466    0.02514 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Commentary

The automatic joint test implemented via `car::linearHypothesis()` yields an F statistic of 2.0466 with a p-value of 0.02538. These values coincide exactly with those obtained from the manual construction of the Wald test in Q2.d, as both procedures rely on the same linear restrictions and the same estimated variance–covariance matrix from the SUR estimator.

([ ] out of 1p) Q2.f: Decision

At the 5% significance level, we reject the joint null hypothesis that all coefficients on `treat` are equal to zero across the system of equations. —

([ ] out of 3p) Q3: Implement Procedure 5 (Hotelling's T2 Test)

Script and Output

```r
X1 <- df[df$treat == 1, opvs]
X0 <- df[df$treat == 0, opvs]

ht <- DescTools::HotellingsT2Test(X1, X0)
ht
```

```
##
##  Hotelling's two sample T2-test
##
## data:  X1 and X0
## T.2 = 2.005, df1 = 10, df2 = 434, p-value = 0.0314
## alternative hypothesis: true location difference is not equal to c(0,0,0,0,0,0,0,0,0,0)
```

```r
str(ht)
```

```
## List of 7
##  $ statistic  : num [1, 1] 2
##   ..- attr(*, "names")= chr "T.2"
##  $ p.value    : num [1, 1] 0.0314
##  $ method     : chr "Hotelling's two sample T2-test"
##  $ parameter  : Named num [1:2] 10 434
##   ..- attr(*, "names")= chr [1:2] "df1" "df2"
##  $ data.name  : chr "X1 and X0"
##  $ alternative: chr "two.sided"
##  $ null.value : Named chr "c(0,0,0,0,0,0,0,0,0,0)"
##   ..- attr(*, "names")= chr "location difference"
##  - attr(*, "class")= chr "htest"
```

```r
# Manual computation (two-sample Hotelling T^2)
n1 <- nrow(X1); n0 <- nrow(X0)
p  <- ncol(X1)

xbar1 <- colMeans(X1)
xbar0 <- colMeans(X0)
dbar  <- matrix(xbar1 - xbar0, ncol = 1)

S1 <- cov(X1)
S0 <- cov(X0)
Sp <- ((n1 - 1) * S1 + (n0 - 1) * S0) / (n1 + n0 - 2)
```

```
T2_manual <- as.numeric((n1 * n0 / (n1 + n0)) * t(dbar) %*% solve(Sp) %*% dbar)

# Finite-sample F transform:
# F = [(n1+n0-p-1) / (p*(n1+n0-2))] * T2  ~  F_{p, n1+n0-p-1}
F_manual <- ((n1 + n0 - p - 1) / (p * (n1 + n0 - 2))) * T2_manual
p_manual <- 1 - pf(F_manual, df1 = p, df2 = n1 + n0 - p - 1)

c(n1=n1, n0=n0, p=p,
  T2_manual=T2_manual,
  F_manual=F_manual,
  p_manual=p_manual)
```

```
##           n1           n0            p    T2_manual     F_manual     p_manual
## 185.00000000 260.00000000  10.00000000  20.46562573   2.00498455   0.03140293
```

```
# Check whether ht$statistic is T^2 or the F-transform
c(ht_stat = as.numeric(ht$statistic),
  close_to_T2 = isTRUE(all.equal(as.numeric(ht$statistic), T2_manual, tolerance = 1e-6)),
  close_to_F  = isTRUE(all.equal(as.numeric(ht$statistic), F_manual, tolerance = 1e-6)))
```

```
##     ht_stat close_to_T2  close_to_F
##    2.004985    0.000000    1.000000
```

Commentary

We implement Procedure 5 by testing the joint null hypothesis that the vector of OPV means is equal across treatment and control groups:

$$H_0 : \ \mathbb{E}[X \mid D = 1] = \mathbb{E}[X \mid D = 0],$$

against the two-sided alternative that at least one component differs.

The reported test statistic is labeled `T.2` but, as verified below, this object equals the finite-sample F-transformation of Hotelling's T^2, not the raw T^2 itself. Specifically, with $p = 10$ OPVs, $n_1 = 185$ treated observations, and $n_0 = 260$ control observations, the raw Hotelling statistic is

$$T^2 = \frac{n_1 n_0}{n_1 + n_0}(\bar{X}_1 - \bar{X}_0)^\top S_p^{-1}(\bar{X}_1 - \bar{X}_0) = 20.466,$$

where S_p is the pooled covariance matrix. The corresponding finite-sample transformation is

$$F = \frac{n_1 + n_0 - p - 1}{p\,(n_1 + n_0 - 2)}\,T^2 \ \sim \ F_{p,\,n_1+n_0-p-1} = F_{10,\,434}.$$

Plugging in the sample values yields

$$F = 2.005, \qquad p\text{-value} = 0.0314.$$

Our code confirms that the statistic returned by `HotellingsT2Test()` coincides numerically with this F-statistic (and not with the raw T^2), as `ht_stat` matches `F_manual` exactly and differs from `T2_manual`. Therefore, the correct interpretation is that the reported value $T.2 = 2.005$ should be read as the $F_{10,434}$ test statistic corresponding to Hotelling's T^2.

At the 5% significance level, we reject H_0. This indicates that the OPVs are not jointly balanced in means between the treated and control groups. This conclusion is consistent with the SUR-based joint Wald/F test in Procedure 4 and the treatment-assignment regression test in Procedure 6, reflecting the theoretical equivalence of these joint balance tests under the companion document's framework.

([ ] out of 4p) Q4: Implement Procedure 6 (OPV do not predict treatment assignment)

Script and Output

```
opvs <- c("age","edu","nodegree","black","hisp",
          "married","u74","u75","re74","re75")

lm_fit <- lm(treat ~ age + edu + nodegree + black + hisp +
               married + u74 + u75 + re74 + re75,
             data = df)

summary(lm_fit)$r.squared
```

```
## [1] 0.04415781
```

```
summary(lm_fit)$fstatistic   # (value, numdf, dendf)
```

```
##      value      numdf      dendf
##   2.004985  10.000000 434.000000
```

```
R2 <- summary(lm_fit)$r.squared
n  <- nobs(lm_fit)
M  <- length(opvs)  # number of slope parameters (10)

F_manual <- (R2 / M) / ((1 - R2) / (n - (M + 1)))
p_manual <- 1 - pf(F_manual, df1 = M, df2 = n - (M + 1))

c(R2 = R2, n = n, M = M, F_manual = F_manual, p_manual = p_manual)
```

```
##           R2            n            M     F_manual     p_manual
##   0.04415781 445.00000000  10.00000000   2.00498455   0.03140293
```

Commentary

We estimate the linear probability model

$$D_i = \alpha + X_i'\beta + v_i,$$

where X_i is the vector of the $J = 10$ OPVs. The hypotheses for Procedure 6 are:

$$H_0:\ \beta_1 = \beta_2 = \cdots = \beta_{10} = 0 \quad \text{vs} \quad H_1:\ \exists k \in 1, \dots, 10 \text{ such that } \beta_k \neq 0.$$

We use the overall F-test (equivalently the R^2-based statistic) for H_0.

([ ] out of 5p) Q5: Implement Procedures 7 and 8 (control for FWER)

Script and Output

```
# Q5: Procedures 7 (Bonferroni) and 8 (Holm-Bonferroni)
# We use the one-at-a-time p-values for H0,j: pi_{1,j}=0 (treat coefficient = 0) for each OPV.

opvs <- c("age","edu","nodegree","black","hisp","married","u74","u75","re74","re75")

# If you already created ols_fits in Q2.b, you can reuse it.
# Otherwise:
ols_fits <- lapply(opvs, function(v) lm(as.formula(paste0(v, " ~ treat")), data = df))
names(ols_fits) <- opvs
```

```r
# Extract unadjusted p-values for treat in each equation
p_raw <- sapply(ols_fits, function(m) summary(m)$coefficients["treat","Pr(>|t|)"])

# Procedure 7: Bonferroni adjusted p-values = p_adj_j = min(1, J * p_j)
# (implemented by p.adjust(method="bonferroni"))
p_bonf <- p.adjust(p_raw, method = "bonferroni")

# Procedure 8: Holm-Bonferroni adjusted p-values (implemented by p.adjust(method="holm"))
p_holm <- p.adjust(p_raw, method = "holm")

# Collect results
res_q5 <- data.frame(
  OPV = opvs,
  p_raw = as.numeric(p_raw),
  p_bonf = as.numeric(p_bonf),
  p_holm = as.numeric(p_holm),
  reject_bonf_5pct = (p_bonf <= 0.05),
  reject_holm_5pct = (p_holm <= 0.05)
)

# Show table sorted by raw p-values (helpful for interpretation)
res_q5[order(res_q5$p_raw), ]
```

```
##               OPV       p_raw     p_bonf     p_holm reject_bonf_5pct
## nodegree nodegree 0.001398352 0.01398352 0.01398352             TRUE
## u75           u75 0.065468962 0.65468962 0.58922066            FALSE
## hisp         hisp 0.076473893 0.76473893 0.61179115            FALSE
## edu           edu 0.135411167 1.00000000 0.94787817            FALSE
## age           age 0.264764269 1.00000000 1.00000000            FALSE
## u74           u74 0.326208987 1.00000000 1.00000000            FALSE
## married   married 0.327408105 1.00000000 1.00000000            FALSE
## re75         re75 0.382253831 1.00000000 1.00000000            FALSE
## black       black 0.649493182 1.00000000 1.00000000            FALSE
## re74         re74 0.982318253 1.00000000 1.00000000            FALSE
##          reject_holm_5pct
## nodegree             TRUE
## u75                 FALSE
## hisp                FALSE
## edu                 FALSE
## age                 FALSE
## u74                 FALSE
## married             FALSE
## re75                FALSE
## black               FALSE
## re74                FALSE
```

Commentary

Q5 controls the family-wise error rate (FWER) when testing balance in the $J = 10$ OPVs using one-at-a-time tests of $H_{0,j} : \pi_{1,j} = 0$ for each OPV j. Following the programming guidance, we implement:

- **Procedure 7 (Bonferroni)**, which adjusts p-values as $p_j^{(B)} = \min\{1, Jp_j\}$; and
- **Procedure 8 (Holm–Bonferroni)**, a step-down procedure that is weakly less conservative than Bonferroni.

Both procedures are implemented using `stats::p.adjust()`. At the 5% significance level, only `nodegree` remains statistically significant after controlling the FWER under both Bonferroni and Holm–Bonferroni adjustments. All other OPVs have adjusted p-values well above 0.05.

Therefore, at $\alpha = \%5$ FWER, we reject $H_{0,j}$ only for OPVs whose adjusted p-values are ≤ 0.05. If only `nodegree` survives, then that is the only OPV you can flag as significant under FWER control.

([   ] out of 50p) PART II: Review of OLS for Causal Analysis

([   ] out of 3p) Q6: Two ways of obtaining the DM estimator

Script and Output

```r
treated  <- read.csv("nswre74_treated.csv")
control  <- read.csv("nswre74_control.csv")
treated$treat <- 1
control$treat <- 0

df <- rbind(treated, control)

df$re74 <- df$re74 / 1000
df$re75 <- df$re75 / 1000

## Keep / check core variables
vars <- c("re78", "treat", "re74", "re75")
stopifnot(all(vars %in% names(df)))

## Optional: quick checks
with(df, table(treat))
```

```
## treat
##   0   1
## 260 185
```

```r
summary(df[, vars])
```

```
##       re78           treat             re74              re75
##  Min.   :    0   Min.   :0.0000   Min.   : 0.0000   Min.   : 0.000
##  1st Qu.:    0   1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.: 0.000
##  Median : 3702   Median :0.0000   Median : 0.0000   Median : 0.000
##  Mean   : 5301   Mean   :0.4157   Mean   : 2.1023   Mean   : 1.377
##  3rd Qu.: 8125   3rd Qu.:1.0000   3rd Qu.: 0.8244   3rd Qu.: 1.221
##  Max.   :60308   Max.   :1.0000   Max.   :39.5707   Max.   :25.142
```

```r
fit1 <- lm(re78 ~ treat, data = df)
rho_ols <- coef(fit1)[["treat"]]

dm <- with(df, mean(re78[treat == 1]) - mean(re78[treat == 0]))

# optional: "manual OLS" slope via covariance/variance
rho_manual <- with(df, cov(treat, re78) / var(treat))

c(rho_ols = rho_ols, dm = dm, rho_manual = rho_manual)
```

```
##    rho_ols         dm rho_manual
##   1794.342   1794.342   1794.342
```

```
all.equal(rho_ols, dm)
```

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

```
all.equal(rho_ols, rho_manual)
```

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

Commentary

Estimate of spec 1 by OLS, regressing post-intervention earnings on the treatment indicator is found by:

$$\texttt{re78}_i = \alpha + \rho\texttt{treat}_i + u_i.$$

Under the random assignment of the NSW experiment, the OLS estimator of ρ coincides with the difference in mean outcomes between treated and control groups. To verify this result, we compute:

1. $\hat{\rho}_{\text{OLS}}$: the OLS slope on $\texttt{treat}_i$;
2. $\widehat{DM} = \overline{\texttt{re78}}_1 - \overline{\texttt{re78}}_0$: the difference in sample average earnings between treated and control;
3. $\hat{\rho}_{\text{manual}}$: the OLS slope computed manually as $\widehat{\text{Cov}}(\texttt{treat}_i, \texttt{re78}_i)/\widehat{\text{Var}}(\texttt{treat}_i)$ (to be sure)

Results:

$$\hat{\rho}_{\text{OLS}} = 1794.342, \qquad \widehat{DM} = 1794.342, \qquad \hat{\rho}_{\text{manual}} = 1794.342.$$

As seen above, all quantities are numerically identical. Indeed, the OLS estimator of the treatment coefficient exactly equals the difference in average 1978 earnings between treated and control units. This was checked by the manual computation in $\hat{\rho}_{\text{manual}}$. The estimated ATE implies that being offered training increases earnings in 1978 by approximately $1,794 on average.

([ ] out of 4p) Q7: Fully-saturated specification in Nodegree

Script and Output

```
df$degree <- 1 - df$nodegree

fit_fs <- lm(re78 ~ 0 + nodegree + degree + treat:nodegree + treat:degree, data = df)
coef(fit_fs)
```

```
##       nodegree         degree nodegree:treat   degree:treat
##       4495.415       4854.493       1154.047       3192.025
```

```
b <- coef(fit_fs)

rho1 <- b["nodegree:treat"]
rho2 <- b["nodegree:treat"]

s1 <- mean(df$nodegree == 1)  # share without degree
s0 <- mean(df$nodegree == 0)  # share with degree

ate_hat <- s1 * rho1 + s0 * rho2

c(
  CATE_nodegree1 = rho1,
  CATE_nodegree0 = rho2,
  ATE = ate_hat
)
```

```
## CATE_nodegree1.nodegree:treat CATE_nodegree0.nodegree:treat
##                         1154.047                        1154.047
##              ATE.nodegree:treat
##                         1154.047
```

Using the fully saturated specification (no constant),

$$re78_i = \alpha_1, nodegree_i + \alpha_2(1 - nodegree_i) + \rho_1(D_i \cdot nodegree_i) + \rho_2\big(D_i \cdot (1 - nodegree_i)\big) + u_i,$$

the CATEs are:

$$\widehat{CATE}(nodegree = 1) = \hat{\rho}_1 = 1154.047, \qquad \widehat{CATE}(nodegree = 0) = \hat{\rho}_2 = 1154.047.$$

Let $\hat{s}_1 = \Pr(nodegree = 1)$ and $\hat{s}_0 = \Pr(nodegree = 0)$ in the sample. Then

$$\widehat{ATE} = \hat{s}_1\hat{\rho}_1 + \hat{s}_0\hat{\rho}_2 = 1154.047$$

which are all numerically equivalent.

([ ] out of 2p) Q8: Get $\widehat{ATE}$ in one step

Implementing the $M = 2$ version of the shortcut specification, we have:

Script and Output

```
B2 <- as.integer(df$nodegree == 1)  # nodegree group
B1 <- 1 - B2                        # degree group

N  <- nrow(df)
N2 <- sum(B2)
N1 <- sum(B1)

Z_eps <- df$treat * B2 * (N2 / N)
Z_varpi1 <- df$treat * (B1 - B2 * (N1 / N2))

fit_short <- lm(df$re78 ~ 0 + B1 + B2 + Z_varpi1 + Z_eps)
coef(fit_short)[["Z_eps"]]
```

```
## [1] 2613.452
```

This specification re-parameterizes the saturated model so that one regressor is a normalized linear combination of the treatment-by-group terms whose coefficient equals the sample-share-weighted average of the group-specific treatment effects. Concretely, with groups B_1, B_2 (degree vs no degree), the OLS normal equations imply that the coefficient on the constructed regressor $Z_{\epsilon,i}$ recovers

$$\hat{\epsilon} = \hat{s}_1\hat{\rho}_1 + \hat{s}_2\hat{\rho}_2 = \widehat{ATE},$$

while the remaining regressors absorb the remaining (orthogonal) variation needed to fit the saturated mean structure.

([ ] out of 22p) Q9: Implications of estimating a not-fully saturated specification in an OPV that takes M distinct values

Let $x_i \in a_1, \ldots, a_M$ be a discrete OPV. Define:

- cell share $s_m = \Pr(x_i = a_m)$ (sample analogue $\hat{s}_m = n_m/n$);

- cell treatment rate $p_m = \Pr(D_i = 1 \mid x_i = a_m)$ (sample analogue $\bar{D}_m = n_{1,m}/n_m$);
- cell treatment effect (CATE) $\rho_m = \mathbb{E}[Y_i(1) - Y_i(0) \mid x_i = a_m]$.

Let $\hat{\rho}_m = \bar{Y}_{1,m} - \bar{Y}_{0,m}$ be the within-cell difference in sample means.

Making use of the shortcut regression:

$$Y_i = \sum_{m=1}^{M} \theta_m \mathbf{1}[x_i = a_m] + \rho D_i + u_i.$$

By the partialling-out theorem,

$$\hat{\rho} = \frac{\sum_i \tilde{D}_i \tilde{Y}_i}{\sum_i \tilde{D}_i^2},$$

where $\tilde{Y}_i = Y_i - \bar{Y}_{m(i)}$ and $\tilde{D}_i = D_i - \bar{D}_{m(i)}$, with $\bar{Y}_m$ and $\bar{D}_m$ denoting cell means.

Work cell-by-cell. For a fixed m,

$$\sum_{i:x_i=a_m} \tilde{D}_i^2 = \sum_{i:x_i=a_m} (D_i - \bar{D}_m)^2 = n_m, \bar{D}_m(1 - \bar{D}_m),$$

because D_i is binary. Also,

$$\sum_{i:x_i=a_m} \tilde{D}_i \tilde{Y}_i = \sum_{i:x_i=a_m} (D_i - \bar{D}_m)(Y_i - \bar{Y}_m) = n_m, \bar{D}_m(1 - \bar{D}_m), (\bar{Y}_{1,m} - \bar{Y}_{0,m}),$$

which follows by expanding the sum separately over treated and controls within cell m.

Therefore,

$$\hat{\rho} = \frac{\sum_{m=1}^{M} n_m \bar{D}_m (1 - \bar{D}_m)(\bar{Y}_{1,m} - \bar{Y}_{0,m})}{\sum_{m=1}^{M} n_m \bar{D}_m (1 - \bar{D}_m)} = \sum_{m=1}^{M} \hat{\omega}_m, \hat{\rho}_m,$$

where

$$\hat{\omega}_m = \frac{\hat{s}_m, \widehat{\mathrm{Var}}(D \mid x = a_m)}{\sum_{j=1}^{M} \hat{s}_j, \widehat{\mathrm{Var}}(D \mid x = a_j)}, \qquad \widehat{\mathrm{Var}}(D \mid x = a_m) = \bar{D}_m(1 - \bar{D}_m).$$

The fully saturated estimator averages cell-specific effects using sample composition:

$$\widehat{ATE} * FS = \sum *m = 1^M \hat{s}_m, \hat{\rho}_m.$$

The shortcut uses variance weights $\hat{\omega}_m$, so $\hat{\rho} = \widehat{ATE}_{FS}$ if either:

1. $\bar{D}_m$ is constant across m, so $\widehat{\mathrm{Var}}(D \mid x = a_m)$ is constant and $\hat{\omega}_m = \hat{s}_m$; or
2. $\hat{\rho}_m$ is constant across m, so any convex weights yield the same average.

Under standard regularity conditions, $\hat{s}_m \to s_m$ and $\bar{D}_m \to p_m$, hence

$$\hat{\omega}_m \xrightarrow{p} \omega_m := \frac{s_m p_m (1 - p_m)}{\sum_{j=1}^{M} s_j p_j (1 - p_j)}.$$

Also $\hat{\rho}_m \to \rho_m$. Therefore

$$\hat{\rho} \xrightarrow{p} \sum_{m=1}^{M} \omega_m \rho_m,$$

a variance-weighted average of CATEs. This equals the population ATE

$$ATE = \sum_{m=1}^{M} s_m \rho_m$$

if and only if either: * p_m is constant in m (treatment rates do not vary with (x)), so $\omega_m = s_m$; or * ρ_m is constant in m (homogeneous effects), so any weighting matches ATE.

If $\hat{s}_m = 1/M$ for all m, then

$$\widehat{ATE} * FS = \frac{1}{M} \sum *m = 1^M \hat{\rho}_m,$$

so the fully saturated estimator is the simple average of the cell-specific differences.

Within cell m, the difference-in-means estimator satisfies (approximately)

$$\mathrm{Var}(\hat{\rho}_m \mid x = a_m) \approx \sigma_m^2 \left(\frac{1}{n_{1,m}} + \frac{1}{n_{0,m}} \right) = \frac{\sigma_m^2}{n_m, \bar{D}_m(1-\bar{D}_m)}.$$

If σ_m^2 is similar across cells, then larger $n_m \bar{D}_m(1-\bar{D}_m)$ corresponds to smaller variance (more precision). The shortcut weight satisfies

$$\hat{\omega}_m \propto \hat{s}_m, \bar{D}_m(1-\bar{D}_m) = \frac{n_m}{n} \bar{D}_m(1-\bar{D}_m),$$

so it places relatively more weight on cells with more within-cell treatment variation (and thus, under comparable noise levels, more precise $\hat{\rho}_m$).

The takeaway is that fully saturating in a discrete OPV averages cell-by-cell treatment effects using cell shares. The shortcut regression (dummies for x, but a common treatment slope) generally targets a different weighted average: it emphasizes cells where treatment assignment has more within-cell variation. The shortcut equals ATE only when treatment rates do not vary across cells (so the weights collapse to cell shares) or when treatment effects are homogeneous across cells.

([ ] out of 14p) Q10: Implement the DM estimator with Regression Adjustment (various specifications)

([ ] out of 3p) Q10.a: Estimate Specifications 2, 3 and 4

([ ] out of 1p) Q10.a.i: Estimate Specification 2

Script and Output

```
df$re74 <- df$re74 / 1000
df$re75 <- df$re75 / 1000

# Spec 2
fit2 <- lm(re78 ~ treat + nodegree + edu, data = df)

# Spec 3 (nodegree + edu + other 8 OPVs; typical NSW OPVs shown)
fit3 <- lm(re78 ~ treat + nodegree + edu + age + black + hisp + married + re74 + re75 + u74 + u75,
           data = df)

# Spec 4: add treat × (age - mean(age))
df$age_c <- df$age - mean(df$age)
fit4 <- lm(re78 ~ treat + nodegree + edu + age + black + hisp + married + re74 + re75 + u74 + u75 +
             treat:age_c,
           data = df)

# ATE estimate + t-test (via lm summary)
summ <- function(fit) summary(fit)$coef["treat", c("Estimate","Std. Error","t value","Pr(>|t|)")]
rbind(Spec2 = summ(fit2), Spec3 = summ(fit3), Spec4 = summ(fit4))
```

```
##       Estimate Std. Error  t value    Pr(>|t|)
## Spec2 1645.927   638.0410 2.579657 0.010212514
## Spec3 1670.709   641.1321 2.605873 0.009479869
## Spec4 1659.602   641.2881 2.587920 0.009981085
```

```
summ(fit2)
```

```
##     Estimate   Std. Error      t value     Pr(>|t|)
## 1.645927e+03 6.380410e+02 2.579657e+00 1.021251e-02
```

Commentary

The table above reports estimates of the ATE of being offered training on 1978 earnings under three regression-adjustment specifications. Across Specifications 2–4, the estimated ATE is remarkably stable, ranging from approximately $1,646 to $1,671. In all cases, the null hypothesis that the ATE is zero is rejected at the 1% level using the standard OLS t-test.

Interpretation: Spec 2 adjusts only for educational attainment (`nodegree`, `edu`). The estimated ATE is positive and statistically significant at the 5% level. This indicates that, even with a minimal set of controls focused on education, the offer of training is associated with an increase in 1978 earnings of roughly $1,646 on average. Because education is a strong predictor of earnings, including it helps account for outcome variation without materially altering the experimental contrast between treated and control units.

([ ] out of 1p) Q10.a.ii: Estimate Specification 3

Script and Output

```
summ(fit3)
```

```
##     Estimate   Std. Error      t value     Pr(>|t|)
## 1.670709e+03 6.411321e+02 2.605873e+00 9.479869e-03
```

Commentary

Spec 3 adds the full set of observed pre-treatment variables to the regression. The estimated ATE remains very close to that from Spec 2 and is again statistically significant at the 1% level. The similarity of the point estimates across Specifications 2 and 3 suggests that additional OPVs beyond education do not meaningfully change the estimated treatment effect, consistent with the plausibility of random assignment. The standard error is also of similar magnitude, indicating modest efficiency gains from the expanded covariate set.

([ ] out of 1p) Q10.a.iii: Estimate Specification 4

Script and Output

```
summ(fit4)
```

```
##     Estimate   Std. Error      t value     Pr(>|t|)
## 1.659602e+03 6.412881e+02 2.587920e+00 9.981085e-03
```

Commentary

Spec 4 allows for treatment-effect heterogeneity by age through an interaction between treatment and centered age. The estimated average effect, evaluated at the sample mean age, is again very close to those obtained under Specifications 2 and 3 and remains statistically significant at conventional levels. This indicates that allowing for age-related heterogeneity does not materially alter the estimated average impact of the training offer, reinforcing the robustness of the ATE estimate across alternative regression-adjustment specifications.

Across all three specifications, the estimated ATE is stable—approximately $1,650–$1,670—and statistically significant. This consistency suggests that regression adjustment using balanced OPVs primarily serves to refine precision rather than to change the estimated treatment effect.

([ ] out of 5p) Q10.b: Reasons to include OPVs when they are balanced

Even if OPVs are balanced between treated and control groups, including them as regression covariates can be useful for at least two reasons:

1. If OPVs explain variation in `re78`, then controlling for them can reduce the residual variance and thereby reduce the standard error of $\hat{\rho}$, yielding higher precision and tighter inference for the ATE. This is relevant even under randomized assignment because balance affects bias, not necessarily variance.
2. Even in experiments, exact balance typically holds only in expectation. In a realized finite sample, adjusting for prognostic covariates can stabilize estimates and improve finite-sample efficiency.

Moving from Spec 2 to Spec 3 adds additional OPVs.

([ ] out of 1p) Q10.c: Is it problematic to regression-adjust for OPVs that are lagged outcomes?

Using lagged outcomes (as is the case for `re74`, `re75`) as regression covariates is not problematic so long as they are measured prior to treatment. In that case they are OPVs (pre-treatment variables) and can help improve precision because they tend to be strong predictors of future earnings. The main concern would be conditioning on variables affected by treatment (post-treatment variables), but `re74` and `re75` are pre-intervention in the NSW setting.

([ ] out of 5p) Q10.d: Interactions of OPV with Treatment Indicator and Two Hypothesis Testing Problems

Script and Output

```r
library(car)

# (i) Test H0: ATE = 0 in Spec 4 (coefficient on treat)
car::linearHypothesis(fit4, "treat = 0")
```

```
##
## Linear hypothesis test:
## treat = 0
##
## Model 1: restricted model
## Model 2: re78 ~ treat + nodegree + edu + age + black + hisp + married +
##     re74 + re75 + u74 + u75 + treat:age_c
##
##   Res.Df        RSS Df Sum of Sq      F   Pr(>F)
## 1    433 1.8634e+10
## 2    432 1.8349e+10  1 284471777 6.6973 0.009981 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# (ii) Test H0: effect does not vary by age (interaction term equals 0)
car::linearHypothesis(fit4, "treat:age_c = 0")
```

```
##
## Linear hypothesis test:
## treat:age_c = 0
##
## Model 1: restricted model
## Model 2: re78 ~ treat + nodegree + edu + age + black + hisp + married +
##     re74 + re75 + u74 + u75 + treat:age_c
##
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
## 1    433 1.8389e+10
## 2    432 1.8349e+10  1  39465057 0.9291 0.3356
```

Commentary

In Specification 4,

$$Y_i = \alpha + \rho D_i + X_i'\beta + \gamma, D_i, (age_i - a\bar{g}e) + u_i.$$

Because age_i is centered, ρ is the average treatment effect evaluated at $age = a\bar{g}e$ (and γ captures linear heterogeneity in age).

For the reported t- and F-tests using `car::linearHypothesis()`, we assume:

1. observations are independent across (i);
2. the regressor matrix has full column rank;
3. the variance estimator used for inference is valid under the maintained error structure

Under URA, the coefficient(s) on D_i (and $D_i \times (age_i - a\bar{g}e)$) have a causal interpretation as sample analogues of average causal effects (and their age-slope heterogeneity) within the maintained linear specification.

([ ] out of 5p) Q11: Mechanisms for NSW intervention to impact post-intervention earnings

The regression-adjustment estimates in Specifications 2–4 are stable and positive:

$$\widehat{ATE} * Spec2 = 1645.927, \quad \widehat{ATE} * Spec3 = 1670.709, \quad \widehat{ATE}_{Spec4} = 1659.602,$$

with p-values around (1%) in each case (Spec2: $p = 0.0102$, Spec3: $p = 0.00948$, Spec4: $p = 0.00998$).

A plausible mechanism consistent with a positive effect of the NSW program on 1978 earnings is that the offer of training increases post-treatment earnings through (i) an extensive-margin channel (higher probability of employment in 1978) and/or (ii) an intensive-margin channel (higher wages and/or hours conditional on employment) via increases in job-specific human capital, work experience, and employability acquired during the subsidized training/employment period. The stability of $\widehat{ATE}$ across alternative covariate sets suggests the result is not driven by functional-form sensitivity to OPVs, but is consistent with a genuine earnings increase attributable to the training offer.