Instrument Strength Test and Effects on Causal Inference

Yuxin Cao (cao224@wisc.edu)

Senior Honors Thesis
Department of Statistics, University of Wisconsin-Madison

May 12, 2023

Abstract

When testing for the causal effect of a treatment on an outcome using instrumental variables, a pre-test is frequently used to assess instrument strength before testing the causal relationship. However, the testing of the causal relationship ignores the strength test done a priori, which can dramatically distort the null distribution of the test statistic for the causal effect. *'conditionalInference'* is an R package that provides an implementation of a method that corrects this distortion. One can use functions in this package to test the causal relationship between a treatment and an outcome after conducting a pre-test using the same data, specifically for the Wald test based on the two-stage least squares estimator and the Anderson-Rubin test.

1 INTRODUCTION

1.1 Motivation: Causal Effect of Education

Suppose that we want to examine whether the level of education has an impact on wages. As Figure 1 shown, there is probably a positive association between education and wages. A common approach is to use statistical hypothesis testing to test this relationship. We postulate a linear relationship between the outcome (i.e. wages) and the explanatory variable (i.e. education), then we can express a simple linear regression model using the equation

$$\text{wages} = \alpha^* + \beta^* \text{educ} + u$$

where wages is the wages, educ is the level of education, α^* is the intercept, β^* is the slope parameter, and u is the error. In this case, we test the hypotheses:

$$H_0 : \beta^* = 0 \text{ vs. } H_1 : \beta^* \neq 0$$

The null hypothesis H_0 indicates that there is no causal effect of education on wages, and the alternative hypothesis H_1 indicates that there is a non-zero causal effect of education on wages. We then collect samples to calculate the relevant t-statistic and the corresponding p-value. Under H_0, the t-statistic for the slope β^* is

$$t = \frac{\hat{\beta}^* - 0}{s.e.(\hat{\beta}^*)}$$

where $\hat{\beta}^*$ is the estimated slope coefficient of regressing wages on education, and $s.e.(\hat{\beta}^*)$ is the standard error of this estimate. If the test is significant, meaning that the p-value is smaller than the significance level, we reject the H_0 and conclude that there is a linear relationship between education and wages.

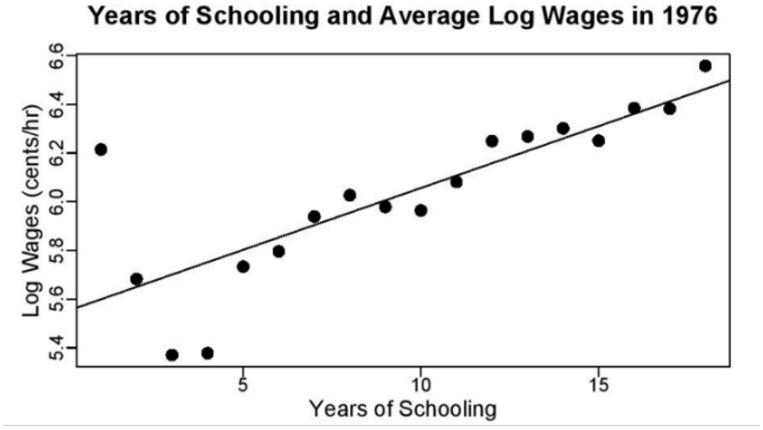

Figure 1: A plot investigating the relationship between average hourly earnings and years of schooling from Rickert (2015). The data was collected by the Bureau of the Census in the 1976 Current Population Survey in the USA. Transforming the outcome wages with the log, an approximately normally distributed variable is obtained, since several economic variables such as wages are assumed to be log-normally distributed. The line in the plot represents the linear model of regressing log wages on years of schooling, and the slope of the line is the treatment effect β^* that we want to estimate.

However, one problem is that the level of education may also be correlated with other variables, such as ability, which is unobserved. As Wooldridge (2013) suggested, this makes the estimates obtained from the ordinary least squares biased and inconsistent. A popular solution is to use instrumental variables. In my honors thesis, I examine the "p-hacking" problem that arises when computing using instrumental variables to test the relationship between education and wages.

1.2 Instrumental Variables (IV)

Instrumental variables (IV) are a popular approach in economics, statistics, and epidemiology to address questions like the above where the regressors may be correlated with unmeasured variables. The IV method uses a special variable called an instrument inside of an estimator, such as the two-stage least-squares estimator, or tests, such as the Anderson-Rubin test (Anderson and Rubin (2019)) and the conditional likelihood ratio test (Moreira (2003)), to assess the causal relationship. The instrument must satisfy the following three core assumptions:

(A1) Relevance: the instrument is correlated with the treatment.

(A2) Validity/Exogenity: the instrument is uncorrelated with confounded variables.

(A3) Exclusion Restriction: the instrument has no direct effect on the outcome.

For the problem of inferring the effect of education and earnings, Angrist and Krueger (1991) suggested using the quarter of birth as an instrument because students who are born in the first quarter of the year enter school at an older age and they can drop out after completing less schooling. Card (1993) also suggested using geographical proximity as an instrument for education because men who grew up in labor markets with a college nearby had a significantly higher education than men who did not grow up near a college. The thesis focuses on the assumption (A1) where we conduct a test for the treatment effect under H_0 after assessing the plausibility of (A1) via an instrument strength test.

1.3 Is the P-Value too Optimistic?

To choose the most appropriate instruments, we often conduct a pre-test on the strength of an instrumental variable (i.e. the level of its association with the treatment). This pre-test will change the null distribution under which we compute p-values. If we use the traditional inference method (see Section 1.1) which ignores testing for instrument strength, our p-value will be the probability of obtaining a result equal to or more extreme than what was actually observed under the null hypothesis that there is no causal effect. In particular, p-values for β^* are computed based on

$$P_{H_0}(T \geq t), t \in \mathbb{R} \tag{1}$$

where the distribution of our test statistic T is assumed to be asymptotically Normal. However, the “honest” p-value will be the probability of obtaining a result equal to or more extreme than what was actually observed condition on the fact that you already know the relationship between instruments and the treatment. That is to say, “honest” p-values are computed based on

$$P_{H_0}(T \geq t | \text{tested IV strength}), t \in \mathbb{R} \tag{2}$$

where the distribution of T is asymptotically a truncated Normal. Ignoring the fact that we use the same data twice, one for testing the instrument strength, and one for inferring the treatment effect, skewed the distribution of the test statistic and inflate Type I error. Our method provides a way to account for the effect of testing for instrument strength a priori and bound the Type I error rate.

2 IV MODEL & DATASET

In this section, we will demonstrate how to develop a linear model with instrumental variables and generate a dataset for testing instrument strength and inferring the treatment effect of this model.

2.1 Two-Stage Linear Model

Consider the following linear model that is commonly used in the IV study (Andrews, Moreira, and Stock (2006); Wooldridge (2013)):

$$\begin{gathered} Y_i = D_i\beta^* + \delta_i \\ D_i = Z_i^T\gamma^* + \xi_{i2} \\ (\delta_i, \xi_{i2}|Z_i) \overset{\text{iid}}{\sim} N(0, \Sigma^*), \quad Z_i \overset{\text{iid}}{\sim} F \end{gathered} \tag{3}$$

where $Y_i \in \mathbb{R}$ denotes the outcome, $D_i \in \mathbb{R}$ denotes the treatment, and $Z_i \in \mathbb{R}^p$ denotes the p instruments for each individual $i = 1, ..., n$. We can form vectors $\mathbf{Y} = (Y_1, ..., Y_n)$ and $\mathbf{D} = (D_1, ..., D_n)$ to represent the outcome and the treatment of all individuals compactly. **Z** will be a *nxp* matrix of instruments, and we assume that matrix **Z** is full rank.

The first line of the equation represents the model to infer the treatment effect after conducting an instrument strength test. The target parameter is β^*, which is the treatment effect and is estimated by the model. The second line of the equation represents the model to test for instrument strength. The parameter γ^*, which is an unknown parameter estimated by the model, stands for the instrument strength. A larger γ value indicates a stronger instrument, while a smaller γ value indicates a weaker instrument. A popular approach is to test the instrument strength via an F-test, which we will explain in detail in Section 3.

2.2 Two-Stage Least-Squares Estimator for Model Parameter

One of the popular and consistent estimators of the treatment effect parameter β^* in the model (3) is the Two-Stage Least-Squares (TSLS) estimator.

Without loss of generality, we assume that **Y**, **D**, **Z** are zero-meaned. By the Frisch–Waugh– Lovell (FWL) theorem, we can reformulate the model in (3). We can obtain the TSLS test statistic in two stages. In the first stage, we establish the instrument by regressing the treatment **D** on the instruments **Z**:

$$\mathbf{D} = \mathbf{Z}^T\gamma^* + \xi, \quad \hat{D} = \mathbf{Z}^T\hat{\gamma}^* \tag{4}$$

where we obtain predicted values $\hat{\mathbf{D}}$. In the second stage, we regress the outcome **Y** on the predicted values $\hat{D}$ obtained in the first stage:

$$\mathbf{Y} = \hat{\mathbf{D}}\beta^* + \delta \tag{5}$$

Under H_0, the estimated TSLS test statistic (i.e. the estimated treatment effect) is

$$\hat{\beta}_{TSLS} = \frac{\mathbf{D}^T P_Z \mathbf{Y}}{\mathbf{D}^T P_Z \mathbf{D}}, \qquad T_{TSLS}(\beta_0) = \frac{\mathbf{D}^T P_Z(\mathbf{Y} - \mathbf{D}\beta_0)}{\sqrt{\hat{\Sigma}_{11}}\sqrt{\mathbf{D}^T P_Z \mathbf{D}}} \tag{6}$$

where $P_Z = \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$ is the projection matrix onto the column space of **Z** and $\hat{\Sigma}_{11}$ is a consistent estimator of Σ^*_{11} (i.e. the true variance of parameter β^*). The left-hand-side equation is the formula for the TSLS estimator, and β_{TSLS} converges to β^* in probability. The right-hand-side equation shows how to derive the TSLS test statistic under H_0.

2.3 Data Generation in R

Before doing any test or inference of the model, we need to obtain a dataset that we can use to estimate the treatment effect. In our R package *'conditionalInference'*, we provide a function called *'bigaussian_instance()'* to generate a dataset, including **Y**, **D**, **Z** matrices, that can later be used to test instrument strength and infer the treatment effect. For example, the following code illustrates how to construct such an instance in R:

```
set.seed(0)
# True beta value is 1
b <- bigaussian_instance(n=1000, p=10, gsnr=1, beta=1,
Sigma = rbind(c(1, 0.8), c(0.8, 1)))

# True beta value is 0
b1 <- bigaussian_instance(n=1000, p=10, gsnr=1, beta=0,
Sigma = rbind(c(1, 0.8), c(0.8, 1)))
```

In the first example, we let the number of samples be 1000, the number of instruments is 10, the true β^* parameter be 1, and the true γ^* parameter be 1. We also specify a $2x2$ variance-covariance matrix for two parameters. In this case, the returned **Y** and **D** are both a $1000x1$ matrix, and **Z** is a $1000x10$ matrix. In addition, the above method also returns the true β^* and a vector that creates a true γ^* for each of the 10 instruments. The dataset can be used for both Two-Stage Least-Squares (TSLS) and Anderson-Rubin (AR) test statistics.

3 METHOD

After constructing the linear model and obtaining the dataset, we are ready to conduct the strength test and develop the inference method. We demonstrate the main techniques of our inference method in this section.

3.1 Testing Instrument Strength with F-test

3.1.1 Mathematical Equation

We start with a pre-test on instrument strength. The most popular pre-test is the F-test. In the first stage of (3), we regress the treatment **D** on instruments **Z** to assess the strength. The null hypothesis $H_0 : \gamma^* = 0$ implies that there is no association between the treatment and instruments, so instruments are irrelevant. The alternative hypothesis $H_1 : \gamma^* \neq 0$ implies that there is an association between the treatment and instruments, so instruments are relevant. The F-statistic is computed by

$$F = \frac{\frac{1}{p}\sum_{i=1}^{n}(\bar{D} - Z_i^T\hat{\gamma})^2}{\frac{1}{n-p}\sum_{i=1}^{n}(D_i - Z_i^T\hat{\gamma})^2} \tag{7}$$

where $\bar{D}$ can be eliminated from the above equation if we re-centered **D** to zero. We need to specify a threshold for the instrument strength test, which is usually $C_0 = 10$. If $F \geq C_0$, then we claim that we have a strong instrument and assumption (A1) is satisfied. We can then proceed to infer the treatment effect, typically with a TSLS test statistic. If $F \leq C_0$, then the instrument is weak, and we can use the AR test statistic to assess the treatment effect.

3.1.2 Testing Instrument Strength in R

In the R package *'conditionalInference,'* there is a method called *'pre_test()'* that can be used to conduct a pre-test on instrument strength:

```
set.seed(0)
Y <- b$Y
D <- b$D
Z <- b$Z
pre_test(Y, D, Z)

## [1] TRUE

pre_test(Y, D, Z, C0=1500)

## [1] FALSE
```

With $\mathbf{Y}$, $\mathbf{D}$, and $\mathbf{Z}$ matrices generated in the previous section, the *'pre_test()'* method performs a F-test for assessing the instrument strength. It returns TRUE if the pre-test is passed (i.e. F-statistic exceeds the threshold) and FALSE otherwise.

Users can specify the threshold as needed. If the threshold is not specified explicitly, the function uses $C_0 = 10$ as a default threshold. In our example, with $C_0 = 10$, $\mathbf{Z}$ is considered a strong instrument. With an arbitrarily large threshold $C_0 = 1500$, $\mathbf{Z}$ is no longer considered a strong instrument. Given that we have strong instruments, the null distribution for test statistics such as TSLS becomes:

$$P_{H_0}(T \geq t | F \geq C_0) \tag{8}$$

If instruments are weak, the null distribution for test statistics such as AR becomes:

$$P_{H_0}(T \geq t | F < C_0) \tag{9}$$

3.2 F-test as a Penalized Convex Optimization

3.2.1 Mathematical Optimization Model

To account for the fact that we have already tested the instrument strength, we first operationalize the conditioning event of $\{F \geq C_0\}$ (in the case of TSLS) as a solution to the penalized convex optimization problem. Adding a randomization term to improve the power of our tests, The randomized version of the problem (see Bi, Kang, Taylor (2020)) becomes

$$\hat{\mathbf{v}} = \underset{\mathbf{v}}{\text{argmin}} \frac{1}{2}||\mathbf{v} - \mathbf{S}||_2^2 + \lambda||\mathbf{v}||_2 - \omega^T \mathbf{v}, \quad \lambda = \sqrt{C_0 \frac{p}{n-p} \left(\sum_{i=1}^{n} (D_i - Z_i^T \hat{\gamma})^2 \right)} \tag{10}$$

where $\mathbf{S} = (\mathbf{Z}^T\mathbf{Z})^{-\frac{1}{2}}\mathbf{Z}^T\mathbf{D} \in \mathbb{R}^p$, λ is the weight of penalty, and ω is a random noise from a known Normal distribution $g(\omega)$ by assumption. Adding random noise to the problem can increase the power and robustness of our tests. The weight of the penalty is calculated based on the sum of squared residuals from the pre-test. $\hat{\gamma}$ is the estimated parameter for instrument strength, and $\mathbf{Z}^T\hat{\gamma}$ gives the predicted $\mathbf{D}$. F-test has numerator degrees of freedom of p and denominator degrees of freedom of $n - p$.

3.2.2 Model Construction in R

After obtaining a dataset consisting of **Y**, **D**, **Z** matrices, we can construct the model in R. Note that generating a dataset using the *'bigaussian_instance()'* method is not a mandatory step to draw the treatment effect. One can also use real-world datasets or their own datasets for this problem. The dataset can be used for both Two-Stage Least-Squares and Anderson-Rubin test statistics.

In our R package, we provide a function called *'group_lasso_iv()'* to construct the model for inferring treatment effects with TSLS test statistics.

```
set.seed(0)
gl <- group_lasso_iv(Y, D, Z, C0=10)
gl1 <- group_lasso_iv(Y, D, Z, penalty=3,
randomizer_scale=0.8, C0=5)

Y1 <- b1$Y
D1 <- b1$D
Z1 <- b1$Z
gl2 <- group_lasso_iv(Y1, D1, Z1, C0=10)
gl$data_part

##           [,1]
##  [1,] 34.57973
##  [2,] 30.67678
##  [3,] 32.47039
##  [4,] 33.76566
##  [5,] 33.35299
##  [6,] 28.84229
##  [7,] 31.30642
##  [8,] 31.00890
##  [9,] 33.60405
## [10,] 29.62992

gl$penalty

## [1] 10.37406
```

This method sets up the penalized convex optimization model in (10). The returned 'data_part' is the $px1$ S matrix and 'penalty' is the λ term. It also returns the log-likelihood of regress 'data_part' on an identity matrix. If 'penalty' is not specified, the method calculates the penalty term based on the equation in (10). The standard deviation of the random noise $\omega = (\omega_1, ..., \omega_n)$ is calculated based on the formula $\{\text{randomizer\_scale} * \text{sd}(\mathbf{S}) * \sqrt{\frac{n}{n-1}}\}$. If 'randomizer_scale' is not specified, we use 'randomizer_scale'=0.5 as the default value.

3.3 Conditional Null Density

3.3.1 Mathematical Model

The next step is to use optimization conditions and change-of-variables to derive the tractable conditional probability of the TSLS test statistic.

Let $\mathbf{u} = \hat{\mathbf{v}}/||\hat{\mathbf{v}}||_2$ with $||\mathbf{u}||_2 = 1$ and $\mathbf{v} = d * \mathbf{u}$ with $d > 0$. Assume that γ^* and Σ^* are known. By the Karush-Kuhn-Tucker (KKT) condition, the solution $\mathbf{v}$ in (10) must satisfy:

$$\omega = \mathbf{v} - \mathbf{S} + \frac{\mathbf{v}}{||\mathbf{v}||_2} = d\mathbf{u} - \mathbf{S} + \lambda\mathbf{u} \tag{11}$$

(see Bi, Kang, and Taylor (2020) for more details). Using changes-of variables, we can reparametrize the conditional null density of the TSLS test statistic as

$$\ell_{\beta_0}(\mathbf{S}, \omega | d > 0, \mathbf{u}) = f_{\beta_0}(\mathbf{S}) * g(d\mathbf{u} - \mathbf{S} + \lambda\mathbf{u}) * \mathbb{I}(d > 0) * |\mathscr{J}|, \quad |\mathscr{J}| = (d+\lambda)^{(p-1)} \tag{12}$$

where $f_{\beta_0}(\mathbf{S})$ is the original null density of $\mathbf{S}$ matrix (i.e. data_part of the model) under H_0. $g(d\mathbf{u} - \mathbf{S} + \lambda\mathbf{u}) = g(\omega)$ is the distribution of the random noise, which is assumed to be Normal. $\mathbb{I}(d > 0)$ is asymptotically equivalent to the conditioning event $\{\mathrm{F} \geq C_0\}$. $\mathscr{J}$ is the Jacobian term from the change-of-variable formula. In other words, the conditional null density is obtained by multiplying the original null density of $\mathbf{S}$ with the effect of passing the F-test and adding randomization to the problem.

Before simulating p-values and confidence intervals, we also need to estimate the variance-covariance matrix. Under $H_0 : \beta^* = \beta_0$, a consistent estimator of the covariance matrix Σ^* in (3) is

$$\hat{\Sigma}(\beta_0) = \frac{1}{n-p} \begin{pmatrix} \mathbf{Y}^T - \mathbf{D}^T\beta_0 \\ \mathbf{D}_T \end{pmatrix} P_{Z\perp} \begin{pmatrix} \mathbf{Y} - \mathbf{D}\beta_0 & \mathbf{D} \end{pmatrix} \tag{13}$$

3.3.2 Deriving the Conditional Null Density in R

The R package *'conditionalInference'* offers a function called *'fit_tsls()'* to establish and obtain results from the penalized convex optimization problem for TSLS test statistic. Input the *'group_lasso_iv'* object obtained in the previous stage to build the model.

```
set.seed(0)
model <- fit_tsls(gl)
model1 <- fit_tsls(gl1)

model$initial_soln

##          [,1]
## [1,] 33.03261
## [2,] 27.55125
## [3,] 28.92558
## [4,] 29.56961
##  [5,] 28.69786
##  [6,] 26.24793
##  [7,] 29.16438
##  [8,] 28.94415
##  [9,] 29.91808
## [10,] 27.65067

model$observed_opt_state

## [1] 91.77022

model$cond_mean

## [1] 90.52298

model$cond_cov

##           [,1]
## [1,] 0.8266297
```

First, the *'fit_tsls()'* function generates an initial value for the randomization term, which is the *'initial_omega.'* It is sampled from the multivariate normal distribution with a mean of zero and a standard deviation of *'randomizer_scale'* (returned by the 'group_lasso_iv()' function).

Then, the function solves the convex optimization problem. The target variables are the p instruments, the objective function is the equation in (10), and we found solutions that minimize the objective function. The solutions are are a $px1$ matrix stored in *'initial_soln,'* and *'observed_opt_state,'* the vector d, is the initial for optimization variables v. We then set up the initial state for optimization variables and compute the conditional mean and covariance.

Additionally, the function makes a box constraint for the truncated Gaussian distribution. As mentioned earlier, the conditioning event $\{F \geq C_0\}$ will make the normal distribution truncated at the lower tail. This box constraint is the core object for affine selection procedures. The function also creates a sampler to generate samples from this truncated Gaussian distribution.

```
initial_omega2 = t(matrix(c(15.10398, 3.426183, 8.380046, 19.18674,
15.99021, -8.367544, 8.134745, -1.295934, -0.8837694, 3.515583)))
model2 <- fit_tsls(gl, perturb = initial_omega2)
model2$observed_opt_state

## [1] 114.3715

model2$cond_mean

## [1] 88.40228
```

```
model2$cond_cov

## [1] 0.8265389
```

Users of this function also have the option to choose their own values for the initial randomization term. Note that amplifying the values of the random noise increases the initial solution for optimization variables but has little impact on the conditional mean and covariance. The null density is robust to the random noise.

In R package *'conditionalInference,'* there is a function called estimate_covariance() that compute the variance-covariance matrix $\hat{\Sigma}(\beta_0)$.

```
set.seed(0)
cov <- estimate_covariance(gl$Y, gl$D, gl$Z)
cov1 <- estimate_covariance(gl1$Y, gl1$D, gl1$Z)

cov

##           [,1]      [,2]
## [1,] 1.0769413 0.8867062
## [2,] 0.8867062 1.0654484
```

According to the above example, with **Y**, **D**, **Z** matrices constructed before, the estimated value for Σ_{11} is $\hat{\Sigma}_{11} = 1.077$, and the estimated value for Σ_{12} is $\hat{\Sigma}_{12} = 0.887$.

3.4 Markov Chain Monte Carlo (MCMC) Sampler

3.4.1 Asymptotic Conditional Null Density

The last step is to use Gibbs sampling to generate samples from the conditional null density of the TSLS test statistic and compute the corresponding p-value and confidence interval.

Bi, Kang, and Taylor (2020) suggest that the conditional density of the TSLS test statistic conditional on passing the F-test, the active direction, and the sufficient statistic **O** as follows

$$\begin{aligned}\ell_{\beta_0}(T_{TSLS}, d|d>0, \mathbf{u}, \mathbf{O}) &= C_T * \phi_{[\beta_0, \widehat{W}_T]}(T_{TSLS}) * g(-\widehat{\mathbf{W}}_{\mathbf{S},T}\widehat{W}_T^{-1} * T_{TSLS} + du + \lambda\mathbf{u} - \mathbf{O}) * |\mathscr{J}| * \mathbb{I}(d>0) \\ &\propto \phi_{[\beta_0, \widehat{W}_T]}(T_{TSLS}) * g(-\widehat{\mathbf{W}}_{\mathbf{S},T}\widehat{W}_T^{-1} * T_{TSLS} + d\mathbf{u} + \lambda\mathbf{u} - \mathbf{O}) * \mathbb{I}(d>0) \\ &= h(T_{TSLS}, d)\end{aligned} \tag{14}$$

where C_T is a normalizing constant and we have

$$\widehat{W}_T = 1, \quad \widehat{\mathbf{W}}_{\mathbf{S},T} = \frac{\widehat{\Sigma}_{12}(\mathbf{Z}^T\mathbf{Z})^{-\frac{1}{2}}\mathbf{Z}^T\mathbf{D}}{\sqrt{\widehat{\Sigma}_{11}\sqrt{\mathbf{D}^T P_Z \mathbf{D}}}} = \frac{\widehat{\Sigma}_{12}\mathbf{S}}{\sqrt{\widehat{\Sigma}_{11}\mathbf{S}^T\mathbf{S}}}, \quad \mathbf{O} = \mathbf{S} - \widehat{\mathbf{W}}_{\mathbf{S},T}\widehat{W}_T^{-1} * \hat{T}_{TSLS} \tag{15}$$

The conditional density in (12) is similar to that in (10), except that (12) is specific for the TSLS test statistic. Also, the conditional null density in (10) is an exact conditional density, which assumes that $f_{\beta_0}(S)$ is known (i.e. model parameters γ^*, Σ^* are known). However, in real-world problems, these model parameters are usually unknown and nuisance parameters. The conditional null density in (12) is an asymptotic conditional null density that gets rid of these nuisance parameters.

3.4.2 Gibbs Sampling

One can then use (12) in an MCMC sampler to generate samples of sufficient statistic $\mathbf{O}$ under H_0 and plug the MCMC samples $\mathbf{O}_1, \ldots, \mathbf{O}_m$ into the TSLS test statistic T_{TSLS} to construct a conditional null distribution of T_{TSLS}.

Consider using a popular MCMC sampling method called Gibbs sampling to generate random samples. Gibbs sampling is particularly useful when it is easier to sample from a conditional distribution than to marginalize over a joint distribution given a multivariate distribution. In this case, we use the density $h(T_{TSLS}, d)$ obtained in (12) to generate samples in the Gibbs sampler.

Let k be the step number and set $k = 0$. Initialize $(T, d) = \{T^{(0)}, d^{(0)}\}$. Assume that we have the k^{th} sample $(T, d) = \{T^{(k)}, d^{(k)}\}$ now. To generate the $(k+1)^{th}$ sample, we follow:

1. Generate a $(k+1)^{th}$ sample for T, denoted by $x^{(k+1)}$, from a normal distribution $N(T^{(k)}, \alpha_k^2)$ (where α_k is the step size). If the sample is accepted, then $x^{(k+1)}$ becomes the $(k+1)^{th}$ sample for T. If not accepted, we retain the previous result $\mathrm{T}^{(k+1)} = T^{(k)}.\mathrm{x}^{(k+1)}$ is accepted with a probability of

$$P(T^{(k+1)} = x^{(k+1)}) = min[\frac{h(x^{(k+1)}, d^{(k)})}{h(T^{(k)}, d^{(k)})}, 1]$$

2. Generate a $(k+1)^{th}$ sample for d, denoted by $y^{(k+1)}$, from a normal distribution $N(d^{(k)}, \beta_k^2)$ (where β_k is the step size). If the sample is accepted, then $y^{(k+1)}$ becomes the $(k+1)^{th}$ sample for d. If not accepted, we retain the previous result $\mathrm{d}^{(k+1)} = d^{(k)}.\mathrm{y}^{(k+1)}$ is accepted with a probability of

$$P(d^{(k+1)} = y^{(k+1)}) = min[\frac{h(T^{(k+1)}, y^{(k+1)})}{h(T^{(k+1)}, d^{(k)})}, 1]$$

3.4.3 Simulating P-Values and Confidence Intervals in R

In our R package, a function called *'summmary_tsls()'* gives summary statistics of hypothesis testing, including pivots, p-values, and confidence intervals of TSLS test statistics.

The two necessary inputs are *'gl,'* which is the 'group_lasso_iv' object obtained in the previous stage, and *'opt_sampler,'* which is the optimization sampler obtained when fitting the model. In the example below, we also let *'Sigma_11'* and *'Sigma_12'* to be estimates in the variance-covariance matrix obtained in (13).

```
set.seed(0)
s <- summary_tsls(gl, model$sampler, Sigma_11 = cov[1,1], Sigma_12 = cov[1,2])
s$observed_target

##           [,1]
## [1,] 0.9877126
```

In this function, we first calculate the estimated treatment effect $\hat{\beta}_{TSLS}$ according to (6). This is the *'observed_target'* attribute in the returned instance, which is approximately equal to 1. The estimated $\hat{\beta}$ is very close to our true β^* parameter, which is set to 1 when constructing our *'bigaussian_instance'* in Section 2.3. There could be multiple observed targets if we want to test a sequence of hypotheses.

```
matrix(s$opt_sample[1:10,1])

##           [,1]
##  [1,] 90.96057
##  [2,] 90.25303
##  [3,] 92.06427
##  [4,] 89.44715
##  [5,] 88.92165
##  [6,] 90.53321
##  [7,] 88.75188
##  [8,] 91.42449
##  [9,] 90.88421
## [10,] 91.97216
```

Since the *'parameter'* argument is not specified (i.e. the β_0 value for null hypothesis), we test the default null hypothesis $H_0 : \beta^* = \beta_0 = 0$. To generate empirical p-values and confidence intervals, we need to first simulate optimization samples from the box constraint that we constructed in Section 3.3.2. We use the Gibbs sampler described in Section 3.4.2 to draw samples of sufficient statistics from a truncated normal distribution with the box constraint, which is stored in the attribute *'opt_sample.'* The output is a $nx1$ matrix; we output just the first 10 samples to save space. One can see that the generated samples are close to the initial state for optimization variables (see Section 3.3.2).

```
s$pivots[[1]]

## [1] 0

s$pvalues[[1]]

## [1] 0
```

Then, we are ready to simulate selective p-values for the TSLS test statistic. We input the observed target statistic, the covariance of the target, the covariance of the target score, and the optimization sampler to the function to calculate the pivot quantities and p-values (see the function *coefficient_pvalues_iv()* for more details).

Pivot quantities are simulated under the null hypotheses $H_0 : \beta^* = \beta_0$, where β_0's are specified by users. We first simulate 'normal' samples from a multivariate normal distribution with mean being the zero matrices and covariance matrix being the average covariances of a sequence of targets (if applicable). We let these ‘normal’ samples be our sample statistics. We compare whether the sample statistics are smaller than the observed statistic, and weigh the results by some weights to compute the pivot quantities.

If all observed target statistics are tested against the null hypothesis $H_0 : \beta^* = 0$, then p-values will be the same as pivot quantities. Otherwise, p-values are computed separately under the null hypothesis $H_0 : \beta^* = 0$ instead of the user-specified hypothesis $H_0 : \beta^* = \beta_0$, where $\beta_0 \neq 0$.

In our example, the true parameter is set to $\beta^* = 1$, and we are testing hypothesis $H_0 : \beta^* = 0$ vs. $H_1 : \beta^* \neq 0$. We assume that there is truly an association between the treatment and the outcome. Both pivot quantities and p-values are 0. This implies that the test is statistically significant, and we have sufficient evidence to reject H_0 and conclude that the treatment has an effect on the outcome. The result is consistent with the β^* value.

```
s$intervals[[1]]

## [1] 0.9720801 1.0053540
```

We can also construct two-sided confidence intervals for the observed TSLS test statistic. We input the observed target statistic, the covariance of the target, the covariance of the target score, the optimization sampler, and the generated optimization samples to the function to construct selective confidence intervals (see the function *confidence_intervals_iv()* and *confidence_interval_iv()* for more details).

We construct a confidence interval for each observed target statistic. The upper limit of the confidence interval is computed by adding an upper ‘offset’ to the observed target. This upper ‘offset’ can be obtained by searching over a grid of values for a root of the function that calculates the pivot quantity of the specified value. The lower limit of the confidence interval can be computed in a similar manner. The default level for confidence intervals is 95%, and users may change the level by specifying a different value for the argument *‘level’* in *summary_tsls()*.

In the example above, the 95% confidence interval contains the true parameter $\beta^* = 1$, so the result is consistent with the calculated pivot quantities and p-values.

```
set.seed(0)
s1 <- summary_tsls(gl1, model1$sampler, Sigma_11 = cov1[1,1], Sigma_12 = cov1[1,2])
s1$observed_target

##            [,1]
## [1,] 0.9877126

s1$pivots[[1]]

## [1] 0

s1$pvalues[[1]]

## [1] 0

s1$intervals[[1]]

## [1] 0.9698168 1.0080680
```

We show another example that test the hypotheses $H_0 : \beta^* = 0$ vs. $H_1 : \beta^* \neq 0$ for a parameter $\beta^* = 0$. One can see that under this example, the observed TSLS test statistic is very small and close to 0. Both the selective pivot quantity and p-value are larger than the significant level $\alpha = 0.05$, so we do not reject H_0, and there is no sufficient evidence to claim that the treatment effect is statistically significant. The selective confidence interval contains the true value $\beta^* = 0$, so it is consistent with the p-value.

```
set.seed(0)
s2 <- summary_tsls(gl, model$sampler, parameter = 1,
Sigma_11 = cov[1,1], Sigma_12 = cov[1,2])
s2$observed_target

##            [,1]
## [1,] 0.9877126

s2$pivots[[1]]

## [1] 0.2089926

s2$pvalues[[1]]

## [1] 0

s2$intervals[[1]]

## [1] 0.9742702 1.0057950
```

Investigators using this function can choose their own *'parameter'* value, which is the β_0 value for the null hypothesis. In the above example, *'parameter'* is set to 1, so we test $H_0 : \beta^* = 1$ vs. $H_1 : \beta^* \neq 1$ for a true value of $\beta^* = 1$. As mentioned earlier, the pivot quantity is computed under the user-specified null hypothesis $H_0 : \beta^* = 1$, which is larger than the 5% significance level and the result is not statistically significant. The p-value is computed under the null hypothesis $H_0 : \beta^* = 0$, so the result rejects the null hypothesis in favor of the alternative hypothesis $H_1 : \beta^* \neq 0$, which is consistent with the true parameter value. The confidence interval also contains the true parameter value.

3.4.4 Comparing Results in Python and R

For validation, we compare the results obtained from our R package *'conditional_Inference'* to that obtained from the implementation in Python. From Table 1, we can see that the results from the R package and the Python version are very similar, which verifies the implementation of our R package.

Package	TSLS	P-value	95%CI
Python	0.996 (0.0001)	0	[0.974 1.016]
R	0.988 (0.0001)	0	[0.972, 1.005]

Table 1: Results from the Python and R packages for TSLS test statistic. The first column represents the estimated TSLS statistic; the second column is the p-value; the third column is the 95% confidence interval.

4 EXTENSIONS

4.1 Weak IV and the AR Test

The asymptotic conditional null in Section 3.4.1 relies on the assumption that our instruments are strong (i.e. the F-statistic for the instrument strength test is above the threshold) However, as mentioned earlier, the F-statistic may be below the threshold, which means that we have weak instruments. We can extend our method to work with the Anderson-Rubin (AR) test, a popular test that is more robust to weak instruments (Anderson and Rubin, 1949). The AR test statistic is denoted as

$$T_{AR}(\beta_0) = \frac{(\mathbf{Y}-\mathbf{D}\beta_0)^T P_Z(\mathbf{Y}-\mathbf{D}\beta_0)/p}{(\mathbf{Y}-\mathbf{D}\beta_0)^T (I-P_Z)(\mathbf{Y}-\mathbf{D}\beta_0)/(n-p)} \tag{16}$$

We test the hypotheses $H_0 : \beta^* = \beta_0$ vs. $H_1 : \beta^* \neq \beta_0$. Under H_0, the AR statistic follows an F distribution with degrees of freedom p and $n-p$.

4.2 Constructing the Model and Deriving the Conditional Null Density in R

After testing for instrument strength, we can perform hypothesis testing of the treatment effect with AR statistics in a similar way to the TSLS statistic. As discussed earlier, the **Y**, **D**, **Z** matrices generated by the 'bigaussian_instance()' function can be used with both TSLS and AR statistics. Thus, we use the same data to test the hypotheses $H_0 : \beta^* = 0$ vs. $H_1 : \beta^* \neq 0$.

```r
set.seed(0)
gl_ar <- group_lasso_iv_ar(Y, D, Z)
model_ar <- fit_ar(gl_ar)
```

We use functions *'group_lasso_iv_ar()'* to construct the model for inferring treatment effects with AR test statistic and *'fit_ar()'* to establish and obtain results from the penalized convex optimization problem for AR test statistic. Note that the procedures for constructing the *'group_lasso_iv'* object, building the optimization model, and obtaining the solution are exactly the same as that in TSLS (see Sections 3.2 and 3.3). We provide different functions to separate usages for TSLS and AR test statistics.

4.3 Simulating P-Values in R

In our R package, the function *'summary_ar()'* gives summary statistics such as p-values for the AR test. The two necessary inputs are *'gl_ar,'* the 'group_lasso_iv_ar' object, and *'opt_sampler,'* the optimization sampler. We also use the same variance-covariance matrix as that in the TSLS case.

```
set.seed(0)
s1_ar <- summary_ar(gl_ar, model_ar$sampler,
                    Sigma_11 = cov[1,1], Sigma_12 = cov[1,2])
s1_ar$observed_target

##             [,1]
##  [1,] 1173.1242
##  [2,]  937.1611
##  [3,] 1026.1742
##  [4,] 1129.4508
##  [5,] 1122.5051
##  [6,]  832.3260
##  [7,]  988.2727
##  [8,]  975.8458
##  [9,] 1129.4385
## [10,]  869.4424
```

We first calculate the observed AR test statistic, represented by the attribute *'observed_target.'* This is a *px*1 matrix, where there is an estimate for each instrument. The AR statistic is computed based on (16).

```
matrix(s1_ar$opt_sample[1:10,1])

##            [,1]
##  [1,] 90.96057
##  [2,] 90.25303
##  [3,] 92.06427
##  [4,] 89.44715
##  [5,] 88.92165
##  [6,] 90.53321
##  [7,] 88.75188
##  [8,] 91.42449
##  [9,] 90.88421
## [10,] 91.97216
```

Similar to that in the TSLS case, we simulate samples of sufficient statistics from the box constraint (see Section 3.3.2) before computing any summary statistics. Therefore, the optimization samples for the AR statistic have similar values to samples simulated for the TSLS statistic.

```
s1_ar$pivots[[1]]

## [1] 0
```

```
s1_ar$pvalues[[1]]

## [1] 0
```

We use the default value 0 for the *'parameter'* argument ($\beta_0 = 0$) since it is not specified. We test the default hypothesis $H_0 : \beta^* = 0$ vs. $H_0 : \beta^* \neq 0$. Note that the AR test rejects only if the observed test statistic is greater than or equal to some threshold (i.e. $AR > k$ for some k), since AR asymptotically follows a F-distribution. But, the alternative hypothesis that the AR is designed to test against is two-sided, which is the same as that for the TSLS statistic.

Pivot quantities and p-values are generated in a similar manner as that in the TSLS case (see Section 3.4.3), except that we compute the test statistic in a different way (see the function *'coefficient_pvalues_iv_ar()'* for more details). Recall that we compare the sample statistic and the observed statistic when calculating pivots/p-values for TSLS statistics. However, for the AR statistic, we do not let 'normal' samples and the *'observed_target'* directly be our sample statistics and observed statistics, respectively. Instead, we use equation (16) to compute them. Specifically, we use 'normal' samples and the *'observed_target'* as β_0 in the numerator to calculate the sample statistic and observed statistic, respectively.

The true parameter is $\beta^* = 1$, while our null hypothesis is $H_0 : \beta^* = 0$. Both pivot quantities and p-values are 0, suggesting that we can reject H_0 in favor of H_1. This implies that the treatment effect is likely to be non-zero, and there is sufficient evidence to conclude that there is a causal effect of the treatment on the outcome. This result is consistent with the fact that we set the true parameter to 1.

5 APPLICATION

For our applications, we re-evaluate the dataset from Card (1993) to compare the results obtained from naive and conditional inference with the TSLS test statistic. As discussed earlier, Card (1993) suggested using geographical proximity to college as an instrument to estimate the effect of years of schooling on earnings. Instruments in this dataset passed the F-test at a threshold of $C_0 = 10$.

5.1 Data Description

Card's dataset consists of $n = 3010$ individuals from the National Longitudinal Survey of Young Men (NLSYM). For the purpose of our application, the data was obtained from *'card.data'* in the R package *'ivmodel'* (Kang, Jiang, Zhao, Small (2021)).

The outcome $\mathbf{Y}$ is the log wages, the treatment $\mathbf{D}$ is education, and the single instrument $\mathbf{Z}$ is a binary indicator of whether a man grew up near a 4-year college. The matrix $\mathbf{X}$ includes 14 other exogenous variables, including years of work experience and its square, whether the individual lived in the South, whether the individual lived in metropolitan areas, whether the individual is black, etc.

As functions in our R package 'conditionalInference' do not take exogenous variables $\mathbf{X}$ as an argument, we need to first rule out the effect of $\mathbf{X}$. We did so by regressing demeaned $\mathbf{Y}$, $\mathbf{D}$, $\mathbf{Z}$ on demeaned $\mathbf{X}$ (i.e. centered these matrices to mean 0), respectively. Denote residuals from these regressions as **rY**, **rD**, **rZ**, and **rY**, **rD**, **rZ** matrices are the data we use for simulation.

5.2 One Instrument Analysis

In this case, we use the geographical proximity to a 4-year college as the single instrument. We use the function *‘pre_test()’* to conduct an F-test for testing the instrument strength with a default threshold of $C_0 = 10$ and verify that the instrument is strong. We use the function *‘naive_inference_tsls()’* to re-run the naive inference method, without accounting for the pre-test on instrument strength, for the TSLS test statistic.

```
result <- pre_test(rY, rD, rZ); result

## [1] TRUE

cov_card <- estimate_covariance(rY, rD, rZ)
naive_inf <- naive_inference_tsls(rY, rD, rZ, pass_pre_test=result,
                                  Sigma_11=cov_card[1,1], compute_intervals=TRUE)
```

For the conditional inference method, we can generate a *‘group_lasso_iv’* object with these residuals matrices, fit the optimization model, estimate the variance-covariance matrix, and simulate conditional pivot quantities, p-values, and confidence intervals as before.

```
set.seed(0)
gl_card <- group_lasso_iv(rY, rD, rZ, C0=10, randomizer_scale=0.0001, perturb = 0)
model_card <- fit_tsls(gl_card)
cov_card <- estimate_covariance(rY, rD, rZ)
s_card <- summary_tsls(gl_card, model_card$sampler, Sigma_11=cov_card[1,1],
                       Sigma_12=cov_card[1,2], ndraw=1000000, burnin=100000)
```

Table 2 below summarizes the results. We test hypotheses $H_0 : \beta^* = 0$ vs. $H_1 : \beta^* \neq 0$. The first row represents the results obtained from *‘naive_inference_tsls()’*, which replicates Card’s original analysis that did not account for the pre-test. The second row represents the results obtained from *‘summary_tsls()’*, which generates summary statistics conditional on passing the instrument strength test. The first column gives the estimated treatment effect with TSLS and standard errors are in parenthesis. The second and third columns show the p-values are 95% confidence intervals.

Method	TSLS	P-value	95%CI
Naive	0.132 (0.055)	0.016	[0.024,0.239]
Conditional	0.132 (0.030)	0.641	[-0.134,0.344]

Table 2: Results from the naive and conditional inference methods using the data from Card (1993).

Both methods estimate a treatment effect of 0.132, which means that holding control variables constant, the model predicts that hourly wages will increase by 14% for every additional year of schooling. However, p-values and 95% confidence intervals differ in the two methods. In naive inference, the p-value is smaller than the 5% significance level, and the 95% confidence interval does not contain $\beta_0 = 0$, so we can reject H_0 and conclude that there is a positive effect of years of schooling on earnings. In contrast, the p-value is greater than the 5% significance level, and the 95% confidence interval does contain $\beta_0 = 0$ in conditional inference. Thus, we do not have sufficient evidence to reject H_0 and cannot suggest that there is an association between years of schooling and earnings.

After conditioning on the pre-test, the treatment effect is no longer significant at the 5% significance level. In other words, ignoring that the same data was used twice, one for testing the instrument's strength and the other to test the treatment effect, led to more optimistic conclusions about the effect of education on earnings and inflated Type I error.

6 CONCLUSION

In this study, we demonstrate how failing to account for the pre-test on instrument strength may significantly skew the distribution of the test statistic and inflate the false positive rate. We use a sampling-based method that controls the false positive rate conditional on passing the pretest and generates corresponding p-values. From my thesis, I learned how to obtain an "honest" inference of the treatment effect after using the same data for a pretest. This method is of Bayesian style, where we update our knowledge given the prior information we have. This conditional inference method is more complicated than the traditional inference method; however, we can obtain a more "honest" inference of the treatment effect.

The current software implementation in R still has some limitations. First, the current R package only includes methods to work with TSLS statistics and AR tests, while other common tests such as the Conditional Likelihood Ratio (CLR) test are not included. When we have weak instruments that do not pass the F-test, the CLR test (Andrews, Moreira, and Stock (2006)) is robust to weak instrument biases and is effective in inferring treatment effects with weak instruments. A future direction to improve the scope of the package is to implement methods for the CLR test so that users can also use the CLR test to infer the treatment effect.

Another limitation is that the confidence intervals for AR statistics can be tricky because there is no simple closed-form formula that allows us to directly compute the confidence interval for AR statistics. The current implementation calculates the pivot quantities/p-values for a grid of statistics $[\widehat{T}_{TSLS} - k, \widehat{T}_{TSLS} + k]$ for some k and computes the confidence intervals by finding the boundary such that p-values become greater than the significance level. However, the current software implementation is not effective or powerful enough to find that boundary as all p-values are smaller than the significance level. This may be because the sampler does not always converge well for AR. Further refinement of the method implementation can provide more informative summary statistics and boost the power of the AR test in this package.

7 ACKNOWLEDGEMENTS

I would like to express my heartfelt thanks to Professor Kang for his guidance and support over the past year as my senior honors thesis advisor. It was a wonderful experience learning and exploring methodology in statistical inference and instrumental variables. We recognize Nan Bi, Hyunseung Kang, and Jonathan

Taylor for providing the novel methodology and code implementation in the paper "Inferring Treatment Effects After Testing Instrument Strength in Linear Models" on which functions in this R package are mainly based. We also want to thank David Card, Hyunseung Kang, Jiang Yang, Qingyuan Zhao, and Dylan S. Small for providing the data set that allows us to experiment with our R package with real-world data.

8 ADDITIONAL DETAILS

The R package *'conditionalInference'* can be downloaded from Github (https://github.com/caoyuxin0406/conditionalInference). The replication R code for this honors thesis is in the Github page under the folder 'demo.' For more details about the usage of a specific function, please refer to its R manual.

REFERENCES

Anderson, Theodore W., and Herman Rubin. "Estimation of the parameters of a single equation in a complete system of stochastic equations." *The Annals of Mathematical Statistics* 20.1 (1949): 46-63.

Andrews, Donald W. K., Marcelo J. Moreira, and James H. Stock. "Optimal Two-Sided Invariant Similar Tests for Instrumental Variables Regression." *Econometrica* 74, no. 3 (2006): 715–52. http://www.jstor.org/stable/4123100.

Angrist, Joshua D., and Alan B. Krueger. "Does Compulsory School Attendance Affect Schooling and Earnings?" *The Quarterly Journal of Economics* 106, no. 4 (1991): 979-1014.

Bi, Nan, Hyunseung Kang, and Jonathan Taylor. "Inference after selecting plausibly valid instruments with application to mendelian randomization." *arXiv preprint arXiv:1911.03985* (2019).

Bi, Nan, Hyunseung Kang, and Jonathan Taylor. "Inferring treatment effects after testing instrument strength in linear models." *arXiv preprint arXiv:2003.06723* (2020).

Card, David. "Using geographic variation in college proximity to estimate the return to schooling." Working Paper 4483, *National Bureau of Economic Research*. (1993).

Kang, Hyunseung, Yang Jiang, Qingyuan Zhao, and Dylan S. Small. "ivmodel: An R Package for Inference and Sensitivity Analysis of Instrumental Variables Models with One Endogenous Variable." *Observational Studies* 7, no. 2 (2021): 1-24. doi:10.1353/obs.2021.0029.

Moreira, Marcelo J. "A conditional likelihood ratio test for structural models." *Econometrica* 71, no. 4 (2003): 1027-1048.

Rickert, Joseph. “Instrumental Variables.” Revolutions, October 29, 2015. https://blog.revolutionanalytics.com/2015/10/instrumental-variables.html.

United States. Bureau of the Census. *Current Population Survey*, June 1976. [distributor], 1992-02-17. https://doi.org/10.3886/ICPSR09282.v1

Wooldridge, Jeffrey M. "Chapter 15: Instrumental Variables Estimation and Two Stage Least Squares." Essay. In *Introductory Econometrics: A Modern Approach*, 5th ed., 512–54. Cengage Learning, 2013.