1 Importing Data Into R

To import csv files into R the first thing to do is to change the working directory in the File drop down menu. Change the working directory to the folder in which the data files are stored. Once this is done, we will use the *read.csv* command. For example suppose we want to import the dataset 401k.csv. To do this we simply enter the following command:

data=read.csv('401k.csv')

The next thing to do is to attach the dataset:

attach(data)

To view the data in spreadsheet form:

fix(data)

2 Basic Statistics in R

To calculate the mean of *all* of the variables in the dataset we can use the commmand *mean*:

mean(data)

We can also calculate the variance-covariance matrix for the dataset using the *var* command:

var(data)

For the standard deviation, while assigning a value to the output, use the *sd* commmand:

sddev< −sd(data)

To get the variance we could also just square the standard deviation:

varr< −sddev∧2

3 Constructing Subsets in R

Suppose we would like to to create a data set which is a subset of the main data file in use. For example, suppose we would like to calculate the average participation rate for only the companies that have a 401k as their sole plan. This corresponds to $sole = 1$ in the data set. We can do this by using the *subset* command:

```
datasub=subset(data,sole==1)
```

After subsetting the data, we can then attach the subset as the main file by first detaching the main data file:

```
detach(data)
```

And then by attaching the subset:

```
attach(datasub)
```

Now calculate the average participation rate:

```
mean(prate)
```

```
[1] 90.07487
```

The average participation rate is higher when $sole = 1$ as compared to the data set as a whole, which has an average prate of 87.36291.

4 Running Regressions in R

To run a simple regression in R *lm* command:

```
output <- lm(prate ~ age + mrate)
```

To summarize the ouput enter the following command:

```
summary(output)
```

Suppose we would just like to run the same regression for only the 401k plans that are older than 8 years old. We can do this by creating a new data set that is a subset of the original one.

```
older8data <- subset(data,age>8)
```

Listing 1: R output

```
Call:
lm(formula = prate ~ age + mrate)

Residuals:
    Min      1Q  Median      3Q     Max
-81.162  -8.067   4.787  12.474  18.256

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  80.1191     0.7790  102.85  < 2e-16 ***
age           0.2432     0.0447    5.44 6.21e-08 ***
mrate         5.5213     0.5259   10.50  < 2e-16 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1

Residual standard error: 15.94 on 1531 degrees of freedom
Multiple R-squared:  0.09225,   Adjusted R-squared:  0.09106
F-statistic: 77.79 on 2 and 1531 DF,  p-value: < 2.2e-16
```

To run the regression on the subset of data you have to first detach the original data set:

detach(data)

Not attach your new data set:

attach(older8data)

Now you are all set to run the regression using only a subset of your original data:

outputolder8 <- lm(prate ~ age + mrate)

summary(outputolder8)

4.1 Partialling Out Approach to Estimation

In this section we will demonstrate the partialling out approach to estimating the multiple linear regression model with the following population model:

$$wage = \beta_0 + \beta_1 educ + \beta_2 IQ + u \tag{1}$$

Listing 2: R output

```
Call:
lm(formula = prate ~ age + mrate)

Residuals:
    Min      1Q  Median      3Q     Max
-79.910  -5.686   4.816   9.831  13.619

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 85.56532    1.22439  69.884  < 2e-16 ***
age          0.07157    0.05356   1.336    0.182
mrate        4.28636    0.60544   7.080 3.24e-12 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1

Residual standard error: 13.51 on 774 degrees of freedom
Multiple R-squared:  0.06483,   Adjusted R-squared:  0.06241
F-statistic: 26.83 on 2 and 774 DF,  p-value: 5.425e-12
```

wagedata=read.csv('wage2.csv')

The next thing to do is to attach the dataset:

attach(wagedata)

To view the data in spreadsheet form:

fix(wagedata)

We can estimate β_1 by running the MLR using the usual lm command:
output1<- lm(wage $\sim$ educ + IQ)
summary(output1)

where $\hat{\beta}_1 = 42.058$.

We can also calculate $\hat{\beta}_1 = \frac{\sum_i^n \hat{r}_{i1} y_i}{\sum_i^n \hat{r}_{i1}^2}$ where $\hat{r}_{i1} = educ_i - \tilde{\delta}_0 - \tilde{\delta}_1 IQ_i$.

The first step is to estimate the regression of *educ* on IQ by using the lm command:
outputwageediq<- lm(educ $\sim$ IQ)

Listing 3: R output

```
Call:
lm(formula = wage ~ educ + IQ)

Residuals:
     Min       1Q   Median       3Q      Max
-860.29 -251.00  -35.31  203.98 2110.38

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -128.8899    92.1823  -1.398    0.162
educ          42.0576     6.5498   6.421 2.15e-10 ***
IQ             5.1380     0.9558   5.375 9.66e-08 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 376.7 on 932 degrees of freedom
Multiple R-squared:  0.1339,    Adjusted R-squared:  0.132
F-statistic: 72.02 on 2 and 932 DF,  p-value: < 2.2e-16
```

summary(outputwageediq)

Listing 4: R output

```
Call:
lm(formula = educ ~ IQ)

Residuals:
     Min       1Q   Median       3Q      Max
-4.7266  -1.4255  -0.3184   1.2786   5.9827

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.846300   0.419127   13.95   <2e-16 ***
IQ          0.075256   0.004093   18.39   <2e-16 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1  1

Residual standard error: 1.883 on 933 degrees of freedom
Multiple R-squared:  0.2659,    Adjusted R-squared:  0.2652
F-statistic:   338 on 1 and 933 DF,  p-value: < 2.2e-16
```

where $\tilde{\delta}_0 = 5.8463$ and $\tilde{\delta}_1 = .07526$.

To calculate $\hat{r}_{i1}$ we can use the residuals command:
rhat=residuals(outputwageediq)

Then we just regress *wage* on our residuals:
outputpart<- lm(wage ~ rhat)

summary(outputpart)

Listing 5: R output

```
Call:
lm(formula = wage ~ rhat)

Residuals:
     Min       1Q   Median       3Q      Max
-848.14 -278.30  -43.07  208.20 2157.85

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  957.945     12.975  73.829  < 2e-16 ***
rhat          42.058      6.898   6.097 1.58e-09 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1     1

Residual standard error: 396.8 on 933 degrees of freedom
Multiple R-squared:  0.03832,    Adjusted R-squared:  0.03729
F-statistic: 37.18 on 1 and 933 DF,  p-value: 1.579e-09
```

This approach produces the same estimate for β_1 as obtained using the MLR approach.

4.2 Testing Other Null Hypotheses

From listing 3 we find that the coefficient estimate for the impact of education on wage is equal to 42.0576. Suppose we are interested in testing whether the *true* impact on wage is actually equal to 75 so that $H_0 : \beta_1 = 75$ versus $H_1 : \beta_1 \neq 75$. To test this we just formulate our test statistic under our H_0 which is $\frac{\hat{\beta}_1 - 75}{se(\hat{\beta}_1)} = \frac{42.0576 - 75}{6.5498} = -5.0295$.

Now that we have our test statistic under the null $H_0 : \beta_1 = 75$, we can just calculate the p-value for the t-stat under a two-sided alternative. To do this, we will use the pt command in R.

pvalue= 2*pt(-5.0295,933)

pvalue

5.89683e-07

Since our pvalue is less than .01 and .05 we reject H_0 in favor of H_1 at the 1% and 5% level.

4.3 Creating Interaction Terms

Suppose we are interested in interacting experience with education. The population model of interest would be as follows:

$$wage = \beta_0 + \beta_1 + \beta_2 exper + \beta_3 exper \times educ + u \tag{2}$$

To do this in R the first thing you want to do is create the new variable which we will call educexper= $educ \times exper$. This is done as follows:
educexper= educ $*$ exper

Next run the regression of wage on education, experience, and the interaction between education and experience.

output1$<-$ lm(wage $\sim$ educ+exper+educexper)
summary(output1)
Which produces the following output:

Listing 6: R output

```
Call:
lm(formula = wage ~ educ + exper + educexper)

Residuals:
    Min      1Q  Median      3Q     Max
-948.46 -254.53  -29.57  192.59 2150.77

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  271.934    230.227   1.181  0.23784
educ          35.106     16.626   2.112  0.03499 *
exper        -32.663     19.099  -1.710  0.08757 .
educexper      3.904      1.462   2.670  0.00771 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 375.1 on 931 degrees of freedom
Multiple R-squared:  0.1424,    Adjusted R-squared:  0.1397
F-statistic: 51.54 on 3 and 931 DF,  p-value: < 2.2e-16
```

The estimate on educexper is the estimate for β_3 from the proposed population model.

4.4 Constructing a F-test for Joint Significance

Suppose we are interested in testing whether mothers education and fathers education is jointly significant in the following model:

$$wage = \beta_0 + \beta_1 + \beta_2 exper + \beta_3 meduc + \beta_4 feduc + u \tag{3}$$

This amounts to proposing the following null hypothesis $H_0 : \beta_3 = \beta_4 = 0$ versus the alternative $H_1 : H_0$ is false. To test this we have to construct our F-statistic with the following formula:

$$F = \frac{(SSR_r - SSR_{ur})/q}{SSR_{ur}/(n-k-1)} \tag{4}$$

where $SSR_r = \sum_{i=1}^{n} \hat{u}_i^2$ from the restricted model and $SSR_u r = \sum_{i=1}^{n} \hat{u}_i^2$ is the sum of the squared residuals from the unrestricted model.

First run the restricted mode and get the residuals:

outputr< − lm(wage ∼ educ+exper)

uhatr= residuals(outputr)

Now create your SSR_r:

SSRr= sum(uhatr ^ 2)

Do the same thing for the unrestricted model:

outputur< − lm(wage ∼ educ+exper+meduc+feduc)

uhatur=residuals(outputur)

Now create your SSR_{ur}:

SSRur=sum(uhatur ^ 2)
Finally create your F-statistic:

F=((SSRr-SSRur)/2)/(SSRur/(717))

F

107.9471

to find the p-value use the pf command:

pvalue=1-pf(F,2,717)

pvalue

0

4.5 Using the robust.r Command

The robust.r command computes the heteroskedasticity robust standard errors as presented in Chapter 8 of Wooldridge.

To use the command you must first create a data maxtix x. To do this just use the cbind.r command as follows:

x< − cbind(educ,exper,meduc,feduc)

Now just run the following command:

robust(wage,x)

Which produces the following output:

Where the robust standard errors are reported at the bottom of the output.

Listing 7: R output

```
Call:
lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-784.04 -256.02  -38.51  203.57 2114.14

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -347.560    122.929  -2.827  0.00482 **
xeduc         62.478      7.601   8.219 9.58e-16 ***
xexper        20.185      3.684   5.479 5.92e-08 ***
xmeduc         9.719      6.152   1.580  0.11461
xfeduc        13.379      5.419   2.469  0.01379 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 376.1 on 717 degrees of freedom
Multiple R-squared:  0.1553,    Adjusted R-squared:  0.1506
F-statistic: 32.95 on 4 and 717 DF,  p-value: < 2.2e-16


$RobustSE
                educ      exper      meduc      feduc
125.372520  7.646293   3.708444   5.323717   5.105549
```