ECON 490 R Homework: *Organizing Data Analysis Output*

In this handout, I'll sketch out several general approaches to including R output in Word files. Then, I'll provide a set of sample tables and figures that you can use as a guide both for completing your R homework assignment and for creating your capstone tables and figures.

General approaches:

1. Typing out results from the R console into Word (simple but time consuming)
2. Exporting objects as an Excel file then copying over to Word (adds a step but less time consuming)
3. Using packages such as `modelsummary` or `stargazer` (lets you automate output but has a steep learning curve)

In the notes below, I will discuss the first two options. The third option, as I've noted above, is helpful when you have a larger project and a lot of output but generally requires a lot of time to get familiar with the specific packages and their options. As a general rule, given the kinds of output most students produce for their capstone projects, I would not recommend this approach unless you are personally interested in learning more about automating output creation.

Typing Out Results

For the example below, we'll consider the first group question from the CPS In-Class R Activity. As part of that question, we explored the relationship between poverty and migration.

To answer this question, I generated an object storing the proportion of people who had moved in the last year across observations who were above and below the poverty line. When I print this object in the console in R, this is what it looks like:

```
> print(migration.proportions)
# A tibble: 2 × 5
  in.poverty total_people moved_people migration_rate poverty_label
       <dbl>        <int>        <dbl>          <dbl> <chr>
1          0       735424        55388         0.0753 Not in Poverty
2          1        97415        11674         0.120  In Poverty
```

The easiest way to include this output in Word is to create a blank table, then type in the values myself. For this output, we might create the following table ourselves using "Insert" > "Table" in Word:

Poverty Level	**Average Migration Rate**
Income Below the Poverty Line	
Income Above the Poverty Line	

After creating the table above in Word by hand, we can simply type out the relevant values from R into our table in Word (while doing some rounding to make the output clear):

Poverty Level	**Average Migration Rate**
Income Below the Poverty Line	12.0%
Income Above the Poverty Line	7.5%

Exporting Objects to Excel then Copying to Word

For shorter pieces of output, typing out your results in a Word table might not be too time consuming. However, for larger output, particularly regression tables with lots of explanatory variables, you may want to find a faster alternative. In the example below, I show you how to clean up regression table output using the `tidy()`, `lm_robust()` and `write_csv()` functions from the `broom`, `estimatr` and `tidyverse` packages. Make sure to load each of these packages before trying these functions!

In the second question from the CPS In-Class R Activity, we explored average wages across various occupations. As part of this activity, we used `lm_robust()` to run a regression of wages on an industry factor variable, corresponding to the 5 most common industries in which people worked. The output from this regression is included below:

```
> summary(wage_model)

Call:
lm_robust(formula = inc.wage ~ as.factor(ind), data = employed_subsample)

Standard error type:  HC2

Coefficients:
                   Estimate Std. Error t value  Pr(>|t|) CI Lower CI Upper    DF
(Intercept)           58229      441.5 131.895 0.000e+00    57364    59095 87852
as.factor(ind)7860    -9656      528.7 -18.262 2.266e-74   -10692    -8620 87852
as.factor(ind)7870     5566      916.3   6.074 1.253e-09     3770     7362 87852
as.factor(ind)8191    18876      970.2  19.457 3.842e-84    16975    20778 87852
as.factor(ind)8680   -30799      567.7 -54.253 0.000e+00   -31912   -29686 87852
```

We can export the results from this object into a .csv (comma-separated values) spreadsheet format that you can then open with Excel. Using the `write_csv()` function in R tells R to output whatever object you've included in parentheses as a new .csv spreadsheet. We can run the following line of code:

```
write_csv(tidy(wage_model), "regression-output.csv")
```

We now open our new .csv file in Excel. It looks like the following:

	A	B	C	D	E	F	G	H	I
1	term	estimate	std.error	statistic	p.value	conf.low	conf.high	df	outcome
2	(Intercept)	58229.32	441.4831	131.8948	0	57364.02	59094.63	87852	inc.wage
3	as.factor(ind)7860	-9656.09	528.7421	-18.2624	2.27E-74	-10692.4	-8619.76	87852	inc.wage
4	as.factor(ind)7870	5565.558	916.304	6.073921	1.25E-09	3769.61	7361.505	87852	inc.wage
5	as.factor(ind)8191	18876.08	970.1539	19.45679	3.84E-84	16974.59	20777.58	87852	inc.wage
6	as.factor(ind)8680	-30799	567.6936	-54.2528	0	-31911.7	-29686.3	87852	inc.wage

Within Excel, you can round numbers, remove scientific notation, etc. to clean up your regression output. Once you've got the numeric values looking the way you want, you can then create a blank table in Word, and copy and paste your numeric output from Excel into Word.

Here's a blank table you might create in Word:

Variable	***Estimate***	***P-Value***
Intercept		
Elementary and Secondary Schools		
Colleges and Universities		
General and Specialty Hospitals		
Restaurants and Other Food Services		

Once you've cleaned up your numeric output, you can copy and paste the values into Word. Note that you can copy and paste across multiple cells in a column by highlighting all the cells at once (that way, you can copy over an entire column instead of going cell-by-cell).

Variable	***Estimate***	***P-Value***
Intercept	58229	0.00
Elementary and Secondary Schools	-9656	0.00
Colleges and Universities	5566	0.00
General and Specialty Hospitals	18876	0.00
Restaurants and Other Food Services	-30799	0.00

Several things to note in the output above:

- No one knows what those industry codes in our R output mean! Adding the industry labels makes things much easier to read.
    - In general, you should ***never*** use R variable names in Word regression tables – instead, create your own labels that will make sense to readers.
- I've rounded the coefficient estimates to the nearest dollar – we don't need decimals since we've got very large values here.
    - In general, avoid showing lots of extra decimals unless they're informative.
- For very small p-values, you can simply round to 2-3 decimal places to indicate that it's some value less than 0.001 or 0.0001.

SAMPLE OUTPUT: *Using the Examples Above*

Table 1: Summary Statistics Table for Migration and Poverty Rates

Poverty Level	**Average Migration Rate**
Income Below the Poverty Line	12.0%
Income Above the Poverty Line	7.5%

NOTE: Table above uses the Current Population Survey (CPS) data to calculate the proportion of people who have moved in the past year based on their family-level poverty status.

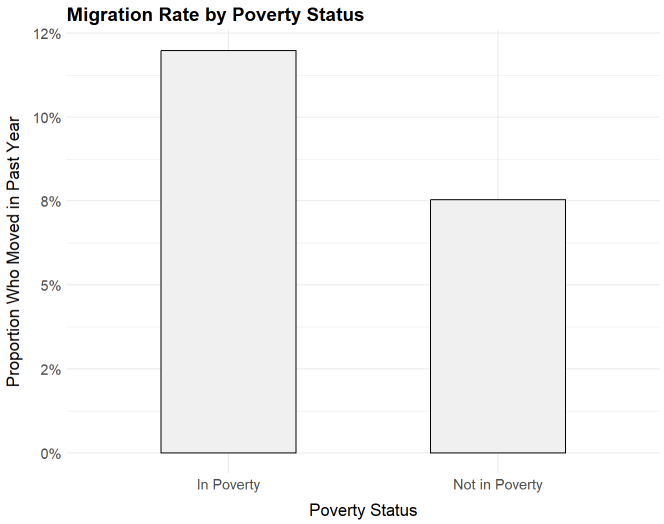
Table 2: Regression Estimates of Differences in Average Wages across Common Employment Industries

Variable	***Estimate***	***P-Value***
Intercept	58229	0.00
Elementary and Secondary Schools	-9656	0.00
Colleges and Universities	5566	0.00
General and Specialty Hospitals	18876	0.00
Restaurants and Other Food Services	-30799	0.00

NOTE: Table above uses the Current Population Survey (CPS) data to estimate average differences in wages across the 5 most common industries of employment. Each industry coefficient above is the coefficient on a binary indicator variable for a person working in that industry. The omitted industry category identified by the intercept term is the construction industry.

NOTE: For your homework, don't forget to include a couple of lines interpreting your regression output!

Figure 1: Barplot Showing the Relationship between Migration and Poverty Rates



NOTE: Figure above uses the Current Population Survey (CPS) data to calculate the proportion of people who have moved in the past year based on their family-level poverty status.