Weiqi Hu
Econ 312 Final Project

Introduction:

In this paper, I explore the properties of OLS coefficient estimators and respective standard deviations, under the violations of SR2, SR3, and SR 4.

These assumptions are described as follows:

- Assumption SR2 maintains that the expected value of the random error *e* is 0.
- Assumption SR 3 maintains that the variance of *e* is he same as the dependent variable *y.*
- Assumption SR 4 maintains that the covariance between any pair of random errors e_i and e_j, or between any pair of y_i and y_j observations is 0.

With my data, I run Monte Carlo simulations to analyze the properties of Ordinary Least Squares (OLS) estimators under my violations.

Data:

The data I use is contained in MC_Class_Demo.dta, which was provided by the instructor. There are 157 observations for the variable x. The following are the summary statistics for x:

```
. summarize

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
           x |       157    1.031624    .1699803       .573      1.384
```

As in Project 1, I will use the following equation for x:

$\square_{\square} = 22 + 6\square_{\square} + \square_{\square}$

I will consider 22 and 6 as the true value of $\square_1$ and $\square_2$ respectively.

Analysis:

OLS with all the assumptions

I start by generating e_i from the normal distribution with mean 0 and standard deviation 0.2. The corresponding do-file is located under Item 1 of the appendix. The coefficient estimates are stored as b1 while the slope coefficient estimates are stored as b2. The do file for running the OLS via Monte Carlo is as follows:

```
program olstest2, rclass
  version 11.1
  g e = rnormal(0,0.2)
  g y = 22+6*x+e
  reg y x
  return scalar b1=_b[_cons]
  return scalar b2=_b[x]
```

```
        drop e
        drop y
      end
```

The next command was:

```
. simulate, reps(10000): olstest2
```

The summaries of the b1 and b2 values are as follows:

```
. summarize b1

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
          b1 |     10000    21.99898    .0985333   21.65528   22.34616

. summarize b2

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
          b2 |     10000    6.000752    .0944802   5.680294   6.308243
```

Note that b1 is the constant estimate and b2 is the slope estimate. As expected, b1 and b2 are close to their true values.

OLS with the error's directly correlated with x

In order to simulate an e value that is correlated with x, I next create e from a normal distribution that takes each corresponding value of x as the variance. This serves as a violation of SR 3, as each e has a different variance which is not necessarily the same as y's variance. The do-file is as follows:

```
program olstest7, rclass
       version 11.1
       scalar u = x
       g e = rnormal(0,u)
       g y = 22+6*x+e
       reg y x
       return scalar b1=_b[_cons]
       return scalar b2=_b[x]
       drop e
       drop y
end
```

The next command was:

```
. simulate, reps(10000): olstest7
```

The summaries of the b1 (_b_cons here) and b2 (_b_x here) values are as follows:

```
. summarize _b_x

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        _b_x |     10000    6.004216    .5295616   4.208048   7.727702

. summarize _b_cons

    Variable |       Obs        Mean    Std. Dev.       Min        Max
```

```
------------+------------------------------------------------------------
     _b_cons |     10000    21.99426     .552279   20.06784    23.9402
```

Both the constant and slope estimates are about .005 further away from the true value for each.

In order to assess heteroskedasticity here (via a Breusch-Pagan test), I would have liked to include a line for it as follows:

```
program olstest7, rclass
       version 11.1
       scalar u = x
       g e = rnormal(0,u)
       g y = 22+6*x+e
       reg y x
       estat hettest // run Breusch-Pagan
       // return hettest values for chi2(1) and Prob>chi2?
       return scalar b1=_b[_cons]
       return scalar b2=_b[x]
       drop e
       drop y
end
```

However, I was unsure how to return the test values.

If I assume that there is some heteroskedasticity is occurring, as e is sampled from a normal curve with varying variance values (u), I can explore the usefulness of attempting to correct for heteroskedasticity via Newey-West robust standard errors. The modified do file is as follows:

```
program olstest23, rclass
       version 11.1
       scalar u = x
       g e = rnormal(0,u)
       g y = 22+6*x+e
       reg y x, robust // now robust
       return scalar b1=_b[_cons]
       return scalar b2=_b[x]
       drop e
       drop y
end
```

The estimates summary is as follows:

```
. summarize b_1

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         b_1 |     10000    21.99914    .5561025   20.16895   24.09074

. summarize b_2

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         b_2 |     10000     5.99981    .5319179   3.972628   7.799638
```

Note that indeed, Newey-West robust standard errors make b_1 and b_2 more accurate, by an order of magnitude.

Autocorrelation

I will be generating errors that that are autocorrelated. Since the error terms are autocorrelated if and only if $\text{cov}(e_t, e_{t-1}) \neq 0$, this is a violation of SR 4.

With first-order serial correlation, the errors in one time period are correlated with errors in the next time period. In effect, errors could also be lagged. This serial correlation does not change the un-biasedness or consistency of OLS estimators, though it does affect efficiency. The OLS estimates of the standard errors will be smaller than the true standard errors.

To explore first-order autocorrelation, I will be generating errors with the following formula:

$$e_t = \rho e_{t-1} + v_t$$

The parameter ρ is the first-order autocorrelation coefficient. We are assuming here that ρ takes any value between -1 and 1. It is effectively the correlation coefficient between e_t and e_{t-1}. Here, I will specifically taking the values of 0.4, 0.8, and 0.99 for ρ. We are assuming that these errors are explanatory and identically distributed with mean zero and a constant variance.

The attempted do file for $\rho = 0.4$ is as follows:

```
program olstes23, rclass
        version 11.1
        gen t = _n
        g u = rnormal(0,0.2)
        replace e = 0.4*e[_n-1] + u
        g y = 22 + 6*x + e
        reg y x, robust // now with robust option
        return scalar b1 = _b[cons]
        return scalar b2 = _b[x]
        drop e
        drop y
end
```

The simulation line was as follows:

```
simulate b_1=r(b1) b_2=r(b2), reps(10000): olstest23
```

Unfortunately, I could not get this to work in stata. The error term did not seem to be recognized.

If I had generated the data, I would test for (first-order) autocorrelation, using using the Durbin-Watson statistic. DW values near 2 indicate no first-order serial correlation, where the range for the statistic is between 0 and 4. I would have additionally tried the Breusch-Godfrey test, which is applicable to autocorrelation of higher orders as well.

Next, I would have investigated how GLS and the Newey-West estimator serve as responses to non-zero autocorrelation.

Afterwards, I would also have explored second-order autocorrelation and tested after simulations. As second-order autocorrelation assumes that e_t is related to the disturbance in period t-1 and t-2, the general formula is given by

$$e_t = \rho_1 e_{t-1} + \rho_2 e_{t-2} + u_t$$

Again, it is assumed that these errors are explanatory and identically distributed with mean zero and a constant variance.

Conclusions and Assessment of Validity:

At least in the case of the error's variance being correlated with x, Newey-West standard errors significantly helped estimates.