33/33, great work!

Assignment two

Question one

The indirect effect of X on Y for Model one is 0.4*0.3=0.12

The direct effect of X on Y is 0.5

The total effect of X on Y is (0.5+0.12) = 0.62.

Question two

```
> # load lavaan
> library(lavaan)
This is lavaan 0.6-3
lavaan is BETA software! Please report any bugs.
Warning message:
package ‘lavaan’ was built under R version 3.5.2
> ## This is lavaan 0.6-3
> ## lavaan is BETA software! Please report any bugs.
> set.seed(123512)
> #
> #
> # Setting up conditions for model 1
> #
> #
> # relationship between X and W
> a <- .4
> # relationship between W and Y
> b <- .3
> # relationship between X and Y
> c <- .5
> # how many observations should we generate.
> # the more we generate the closer we will get to the true parameters
> # (i.e, the small error we'll have) recall sampling distributions.
> n <- 500
> # let's generate X to be a random normal variable
> X <- rnorm(n = n, mean = 0, sd = 1)
> # let's generate W
> W <- a*X + rnorm(n, mean = 0, sd = sqrt(1 - a^2))
> # - The .4 is the standardized regression weight between X and W
> # - The rnorm() stuff adds the residual variance to make the correlation be
tween
> # - X and W .4
> # - You can verify this by setting n to a really large number and doing
> # - cor(X, W)
> # Now let's generate Y
> Y <- b*W + c*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + c^2 + 2*b*c*a)))
> dat <- data.frame(X, W, Y)
> # And let's fit a path model and omit W
> mod <- '
+ Y ~ X
+ '
> fit <- sem(model = mod, data = dat)
```

```
> summary(fit)
lavaan 0.6-3 ended normally after 11 iterations

  Optimization method                           NLMINB
  Number of free parameters                          2

  Number of observations                           500

  Estimator                                         ML
  Model Fit Test Statistic                       0.000
  Degrees of freedom                                 0
  Minimum Function Value               0.0000000000000

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  Y ~
    X                 0.590    0.032   18.446    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Y                 0.574    0.036   15.811    0.000

> params <- parameterEstimates(fit)
> params[params$lhs == "Y" & params$rhs == "X", "est"]
[1] 0.5895919
```

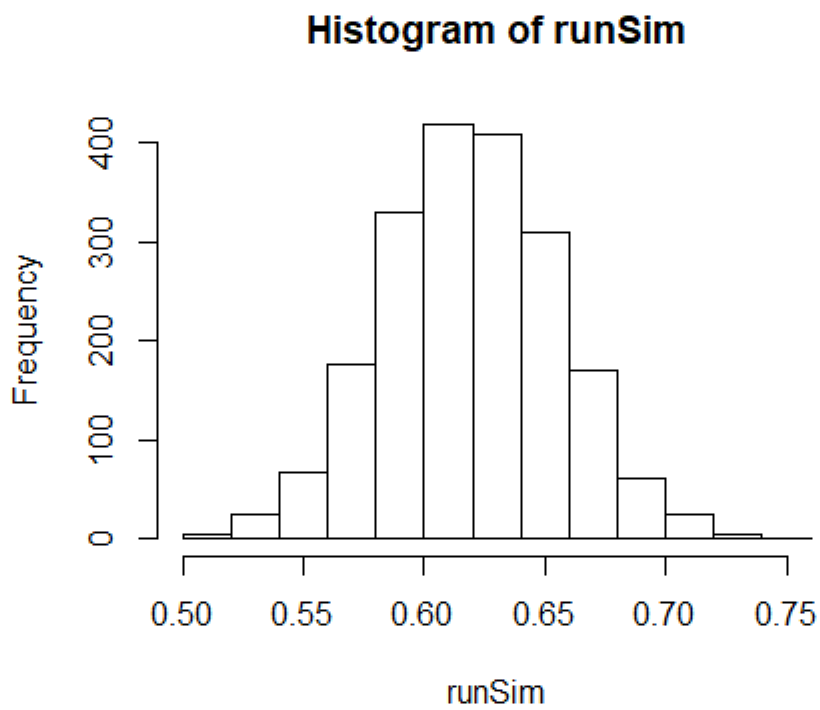
The estimated effect was 0.59. This was very close to the value what I have calculated in Question one. The difference is only (0.62-0.59) = 0.03. I would like to call this a small difference.

Question three

Describing the distribution of *runSim*

```
> set.seed(125312)
> # how many replicates should we use?
> nsim <- 2000
> # run the simulation
> # notice that everything in the expr argument was defined earlier.
> runSim <- replicate(nsim, expr = {
+     # all these lines are the same as above
+     X <- rnorm(n = n, mean = 0, sd = 1)
+     W <- a*X + rnorm(n, mean = 0, sd = sqrt(1 - a^2))
+     Y <- b*W + c*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + c^2 + 2*b*c*
a)))
+     dat <- data.frame(X, W, Y)
+     fit <- sem(model = mod, data = dat)
+ params <- parameterEstimates(fit)
+ params[params$lhs == "Y" & params$rhs == "X", "est"]
```

```
+ })
> # let's plot the results
> hist(runSim)
> # Now calculate the mean
> mean(runSim)
[1] 0.6192045
> hist(runSim)
> mean(runSim)
[1] 0.6192045
```



Distribution of *runSim*

The histogram of runSim indicates that data seems rougly normally distributed. This histogram signifies that there are some outliers in the beginning and end of the data. However, there are more outliers in the end of the data than the beginning which makes the distribution slightly skewed to the right or slightly positive skewness.

Wow,I don't see that. You have a very careful eye :-)

Question 4

The mean of our simulation is 0.619 or 0.62. It indicates that there is basically no difference or a very small difference (0.001) with the total effect of X on Y. If there is a difference between the mean of simulation and the total effect of X on Y, we could say that the total effect is biased. As there is no difference, we could say that the total effect is unbiased when W is omitted.

Question 5

The total effect of X on Y is (0.4)*(0.3) = 0.12

The indirect effect of X on Y is also same that is 0.12

However, there is no direct effect of X on Y that is 0.

Question 6

```
set.seed(125312)
> #
> #
> # Setting up conditions for model 2
> #
> #
> #
> # Change one of these parameters
> #
> # relationship between X and W
> a <- .4
> # relationship between W and Y
> b <- .3
> # relationship between X and Y
> c <- 0
> #
> #
> #
> # how many observations should we generate.
> # the more we generate the closer we will get to the true parameters
> # (i.e, the small error we'll have) recall sampling distributions.
> n <- 500
> # how many replicates should we use?
> nsim <- 2000
> # run the simulation
> # notice that everything in the expr argument was defined earlier
> runSim <- replicate(nsim, expr = {
+     # all these lines are the same as above
+     X <- rnorm(n = n, mean = 0, sd = 1)
+     W <- a*X + rnorm(n, mean = 0, sd = sqrt(1 - a^2))
+     Y <- b*W + c*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + c^2 + 2*b*c*
a)))
+     dat <- data.frame(X, W, Y)
+     fit <- sem(model = mod, data = dat)
+     params <- parameterEstimates(fit)
+     params[params$lhs == "Y" & params$rhs == "X", "est"]
+ })
> # let's plot the results
> hist(runSim)
> # Now calculate the mean
> mean(runSim)
[1] 0.1189428
```

The mean from the simulation is 0.12 or 0.118. It indicates that there is basically no difference (very small) because the total effect of X on Y is 0.12. If there is a difference between the mean of simulation and the total effect of X on Y, we could say that the total effect is biased. At this moment, the total effect is unbiased. Therefore, it indicates that we do not need W in the model to obtain an unbiased estimate of the total effect of X on Y for this model.

Question 7

The total effect of X on Y is 0.5. The direct effect of X on Y is also 0.5. There is no indirect effect of X on Y that is zero. However, W has indirect effect on Y and this is (0.4*0.5) =0.20.

Overachiever ;-)

Question 8

```
set.seed(125312)
> #
> #
> # Setting up conditions for model 3
> #
> #
> #
> # Change one or more of these (if necessary)
> #
> # relationship between W and X
> a <- .4
>
> # relationship between W and Y
> b <- 0
> # relationship between X and Y
> c <- .5
> #
> #
> #
> # how many observations should we generate.
> # the more we generate the closer we will get to the true parameters
> # (i.e, the small error we'll have) recall sampling distributions.
> n <- 500
> # how many replicates should we use?
> nsim <- 2000
> # run the simulation
> runSim <- replicate(nsim, expr = {
+     # all these lines are the same as above
+     W <- rnorm(n = n, mean = 0, sd = 1)
+     X <- a*W + rnorm(n, mean = 0, sd = sqrt(1 - a^2))
+     Y <- b*W + c*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + c^2 + 2*b*c*
a)))
+     dat <- data.frame(X, W, Y)
+     fit <- sem(model = mod, data = dat)
+     params <- parameterEstimates(fit)
+     params[params$lhs == "Y" & params$rhs == "X", "est"]
+ })
> # let's plot the results
> hist(runSim)
> # Now calculate the mean
> mean(runSim)
[1] 0.4994052
```

The mean from the simulation is 0.50 or 0.499. It indicates that there is basically no difference (very small) because the total effect of X on Y is 0.50. At this moment, the total effect is unbiased. Therefore, it indicates that we do not need W in the model to obtain an unbiased estimate of the total effect of X on Y for this model.

Question 9

The total effect of X on Y is 0.5. The direct effect of X on Y is also same and that is 0.5. There is no indirect effect of X on Y and that is zero. W has both direct and indirect effects on Y. The direct effect of W on Y is 0.3 and indirect effect of W on Y is 0.4*0.5=0.20.

Question 10

```
> set.seed(125312)
> #
> #
> # Setting up conditions for model 4
> #
> #
> #
> # Change one or more of these (if necessary)
> #
> # relationship between W and X
> a <- .4
> # relationship between W and Y
> b <- .3
> # relationship between X and Y
> c <- .5
> #
> #
> #
> # how many observations should we generate.
> # the more we generate the closer we will get to the true parameters
> # (i.e, the small error we'll have) recall sampling distributions.
> n <- 500
> # how many replicates should we use?
> nsim <- 2000
> # run the simulation
> # notice that everything in the expr argument was defined earlier.
> runSim <- replicate(nsim, expr = {
+     # all these lines are the same as above
+     W <- rnorm(n = n, mean = 0, sd = 1)
+     X <- a*W + rnorm(n, mean = 0, sd = sqrt(1 - a^2))
+     Y <- b*W + c*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + c^2 + 2*b*c*
a)))
+     dat <- data.frame(X, W, Y)
+     fit <- sem(model = mod, data = dat)
+     params <- parameterEstimates(fit)
+     params[params$lhs == "Y" & params$rhs == "X", "est"]
+ })
> # let's plot the results
> hist(runSim)
> # Now calculate the mean
> mean(runSim)
[1] 0.6192396
```

The mean from the simulation is 0.62. However, the total effect of X on Y is 0.5. We could find t hat the mean from the simulation differs from the total effect of X on Y, it provides a biased effe ct. Therefore, we need W in the model to obtain an unbiased estimate of the total effect of X on Y.

Question 11

The total effect of X on Y is 0.5. The direct effect of X on Y is also 0.5. However, there is no ind irect effect of X on Y that is 0. X has an indirect effect on W and that one is 0.15 and X has a dir ect effect on W and that one is 0.4

Question 12

```
> set.seed(125312)
> #
> #
> # Setting up conditions for model 5
> #
> #
> # relationship between X and W
> a <- .4
>
> # relationship between W and Y
> b <- .3
> # relationship between X and Y
> c <- .5
> #
> #
> #
> # how many observations should we generate.
> # the more we generate the closer we will get to the true parameters
> # (i.e, the small error we'll have) recall sampling distributions.
> n <- 500
> set.seed(125312)
> # how many replicates should we use?
> nsim <- 2000
> # create the misspecified model - this one includes W
> mod.w <- '
+ Y ~ X + W
+ '
> # run the simulation
> # notice that everything in the expr argument was defined earlier.
> runSim <- replicate(nsim, expr = {
+     # all these lines are the same as above
+     X <- rnorm(n = n, mean = 0, sd = 1)
+     Y <- c*X + rnorm(n, mean = 0, sd = sqrt(1 - c^2))
+     W <- b*Y + a*X + rnorm(n, mean = 0, sd = sqrt(1 - (b^2 + a^2 + 2*b*c*
a)))
+     dat <- data.frame(X, W, Y)
+     fit <- sem(model = mod, data = dat)
+     fit.w <- sem(model = mod.w, data = dat)
+     params <- parameterEstimates(fit)
```

```
+     params.w <- parameterEstimates(fit.w)
+     c(params[params$lhs == "Y" & params$rhs == "X", "est"],
+       params.w[params.w$lhs == "Y" & params.w$rhs == "X", "est"])
+ })
> # Now print the means
> rownames(runSim) <- c("Correct", "Misspecified")
> rowMeans(runSim)
    Correct Misspecified
  0.5002590    0.3234937
```

The mean from the simulation for the correct model is 0.5. This is an unbiased effect of X on Y because the total effect of X on Y is 0.5. The mean from the simulation for the Misspecified model is 0.32. This is not an unbiased estimate. This is a biased estimate of X on Y because the mean from the simulation from the misspecified model differs from the total effect of X on Y.

Question 13

It is fine to omit W for Model 1, Model 2, Model 3, and correct model which is obtained through the simulation process for Model 5 to get an unbiased effect. We must include W for Model 4 to get an unbiased effect. For the Misspecified model which is obtained through the simulation process for Model 5, we make sure not to include W.