Roy Model

```
# We will load the tidyverse, a commonly used set of R libraries
# Find more information here: https://www.tidyverse.org/packages/
library('tidyverse')
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library('MASS')
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

Roy Model Simulation 1

Generate Random Data

Economist earnings are normally distributed with mean = 60k and stdev = 10k, and Accountant earnings are normally distributed with mean = 65k and st dev = 5k. This generation will produce data that pairs draws from both distribution, i.e. we are observing for every theoretical individual in our sample both their potential earnings as an accountant and their potential earnings as an economist.

```
set.seed(10042018)

samples <- 1000000  # Sample size or size of population
cor <- 0.9 # correlation coefficient

# Generate correlated std normal random sample using the mvnorm function from MASS package
data <- mvrnorm(n=samples, mu=c(0, 0), Sigma=matrix(c(1, cor, cor, 1), nrow=2), empirical=FALSE)

# Convert to dataframe
df1 <- as_data_frame(data)
```

```
## Warning: ‘as_data_frame()‘ was deprecated in tibble 2.0.0.
## i Please use ‘as_tibble()‘ (with slightly different semantics) to convert to a
##   tibble, or ‘as.data.frame()‘ to convert to a data frame.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
## Warning: The ‘x‘ argument of ‘as_tibble.matrix()‘ must have unique column names if
## ‘.name_repair‘ is omitted as of tibble 2.0.0.
## i Using compatibility ‘.name_repair‘.
## i The deprecated feature was likely used in the tibble package.
##   Please report the issue at <https://github.com/tidyverse/tibble/issues>.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```r
# Make the dataframe easier to use
df1 <- df1 %>% dplyr::rename(accnt = V1, econ = V2) # rename columns

# Change distributions from std normal to those specified in lecture:
# Accounting ~ N(65000, 5000)
# Economics ~ N(60000, 10000)
mu_econ <- 60000
sigma_econ <- 10000
mu_accnt <- 65000
sigma_accnt <- 5000

df1 <- df1 %>% mutate(
        accnt = accnt*sigma_accnt + mu_accnt, # update accounting variable
        econ = econ*sigma_econ + mu_econ # update econ variable
)
```

Sanity Checks

Let’s run a few checks to make sure data looks reasonable. First we’ll use the head and tail commands to get a look at the data. Then, we will make sure that the correlation between the accoutant and economist distributions is 0.86, the value we set when we generated the data, using the cor command. Finally, we will run the summarise command to see the mean, min, max, and quartiles of each of the distributions.

```r
# Look at the first and last 6 rows of the dataframe
head(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 65037. 58736.
## 2 66280. 65719.
## 3 64605. 61839.
## 4 65908. 63792.
## 5 58921. 49206.
## 6 63169. 48734.
```

```r
tail(df1)
```

```
## # A tibble: 6 x 2
##     accnt   econ
##     <dbl>  <dbl>
## 1 58520. 43816.
## 2 68335. 64071.
## 3 62921. 54838.
## 4 61847. 51383.
## 5 70867. 69141.
## 6 68559. 68678.
```

```r
# Compare the correlation we set to the correlation we calculate
check <- round(cor(df1$accnt, df1$econ)) == round(cor)
print(ifelse(check, "The correlations are the same!", "Oops, the correlations are not the same."))
```

```
## [1] "The correlations are the same!"
```

```r
# Generate summary statistics
print("Economists Summary Stats")
```

```
## [1] "Economists Summary Stats"
```

```r
summary(df1$econ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11271   53253   59997   59997   66726  106422
```

```r
print("Accountant Summary Stats")
```

```
## [1] "Accountant Summary Stats"
```

```r
summary(df1$accnt)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   40451   61637   64999   65002   68365   90649
```

```r
### Picking a Career

#Our assumption is that every person will choose to be an accountant or an
#economist based on what will maximize their salary. We will assign job labels
#strictly based on where an individual will earn more,
#i.e. Y<sub>i</sub> = max(Y<sub>e,i</sub>, Y<sub>a,i</sub>)
#```{r jobs}
# Assign job labels
df1 <- df1 %>% mutate(job = ifelse(econ > accnt, "econ", "accnt"))

# Take a look at the change to the dataframe using head
head(df1)
```

```
## # A tibble: 6 x 3
##    accnt   econ job
##    <dbl>  <dbl> <chr>
## 1 65037. 58736. accnt
## 2 66280. 65719. accnt
## 3 64605. 61839. accnt
## 4 65908. 63792. accnt
## 5 58921. 49206. accnt
## 6 63169. 48734. accnt
```

Results

```
# Make results dataframe
results <- df1 %>%
            group_by(job) %>% # Group all of the rows with the same "job" together
            summarise('Economist Earnings' = mean(econ), 'Accountant Earnings' = mean(accnt),

            n=n()) %>% # Calculate means and counts for economicts and accountants
            mutate(job = c("Accountant", "Economist")) %>% # add labels
            t() # transpose

colnames(results) <- c('Accountant', 'Economist')




results <- as.data.frame(results) %>%
            slice(2:4) %>%
            mutate(
              x= c('Economist Earnings', 'Accountant Earnings', 'n')
            ) %>%
            dplyr::select(x, Accountant, Economist)

results
```

```
##                                        x Accountant Economist
## Economist Earnings   Economist Earnings   56756.85  73066.65
## Accountant Earnings Accountant Earnings   63822.98  69759.32
## n                                      n     801309    198691
```

Roy Model Simulation 2

Generate Random Data

Example: Economist earnings are normally distributed with mean = 60k and stdev = 10k, and Accountant earnings are normally distributed with mean = 65k and st dev = 5k. This generation will produce data that pairs draws from both distribution, i.e. we are observing for every theoretical individual in our sample both their potential earnings as an accountant and their potential earnings as an economist.

```
set.seed(02101870)
```

```r
samples <- 1000000  # Sample size or size of population
cor <- 0.9 # correlation coefficient

# Generate correlated std normal random sample using the mvnorm function from MASS package
data <- mvrnorm(n=samples, mu=c(0, 0), Sigma=matrix(c(1, cor, cor, 1), nrow=2), empirical=FALSE)

# Convert to dataframe
df1 <- as_data_frame(data)

# Make the dataframe easier to use
df1 <- df1 %>% dplyr::rename(accnt = V1, econ = V2) # rename columns

# Change distributions from std normal to those specified in lecture:
# Accounting ~ N(65000, 5000)
# Economics ~ N(60000, 10000)
mu_econ <- 60000
sigma_econ <- 10000
mu_accnt <- 65000
sigma_accnt <- 5000

df1 <- df1 %>% mutate(
        accnt = accnt*sigma_accnt + mu_accnt, # update accounting variable
        econ = econ*sigma_econ + mu_econ # update econ variable
)
```

Sanity Checks

Let's run a few checks to make sure data looks reasonable. First we'll use the head and tail commands to get a look at the data. Then, we will make sure that the correlation between the accoutant and economist distributions is 0.86, the value we set when we generated the data, using the cor command. Finally, we will run the summarise command to see the mean, min, max, and quartiles of each of the distributions.

```r
# Look at the first and last 6 rows of the dataframe
head(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 66606. 63314.
## 2 65084. 63758.
## 3 70090. 65585.
## 4 56853. 50869.
## 5 60755. 53398.
## 6 73071. 80320.
```

```r
tail(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 74382. 75437.
## 2 59409. 55834.
## 3 67347. 56298.
## 4 58116. 40943.
## 5 64621. 65722.
## 6 59699. 56242.
```

```
# Compare the correlation we set to the correlation we calculate
check <- round(cor(df1$accnt, df1$econ)) == round(cor)
print(ifelse(check, "The correlations are the same!", "Oops, the correlations are not the same."))
```

```
## [1] "The correlations are the same!"
```

```
# Generate summary statistics
print("Economists Summary Stats")
```

```
## [1] "Economists Summary Stats"
```

```
summary(df1$econ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15650   53234   59972   59988   66729  107892
```

```
print("Accountant Summary Stats")
```

```
## [1] "Accountant Summary Stats"
```

```
summary(df1$accnt)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   41577   61628   64995   64999   68367   89273
```

```
### Picking a Career

#Our assumption is that every person will choose to be an accountant or an
#economist based on what will maximize their salary. We will assign job labels
#strictly based on where an individual will earn more,
#i.e. Y<sub>i</sub> = max(Y<sub>e,i</sub>, Y<sub>a,i</sub>)
#```{r jobs}
# Assign job labels
df1 <- df1 %>% mutate(job = ifelse(econ > accnt, "econ", "accnt"))

# Take a look at the change to the dataframe using head
head(df1)
```

```
## # A tibble: 6 x 3
##    accnt   econ job
##    <dbl>  <dbl> <chr>
## 1 66606. 63314. accnt
## 2 65084. 63758. accnt
## 3 70090. 65585. accnt
## 4 56853. 50869. accnt
## 5 60755. 53398. accnt
## 6 73071. 80320. econ
```

Results

```
# Make results dataframe
results <- df1 %>%
            group_by(job) %>% # Group all of the rows with the same "job" together
            summarise('Economist Earnings' = mean(econ), 'Accountant Earnings' = mean(accnt),
            n=n()) %>% # Calculate means and counts for economicts and accountants
            mutate(job = c("Accountant", "Economist")) %>% # add labels
            t() # transpose

colnames(results) <- c('Accountant', 'Economist')




results <- as.data.frame(results) %>%
            slice(2:4) %>%
            mutate(
              x= c('Economist Earnings', 'Accountant Earnings', 'n')
            ) %>%
            dplyr::select(x, Accountant, Economist)

results
```

```
##                                         x Accountant Economist
## Economist Earnings   Economist Earnings   56745.38  73043.11
## Accountant Earnings Accountant Earnings   63819.94  69745.89
## n                                       n     801019    198981
```

Roy Model Simulation 3

Generate Random Data

Example : Economist earnings are normally distributed with mean = 60k and stdev = 10k, and Accountant earnings are normally distributed with mean = 65k and st dev = 5k. This generation will produce data that pairs draws from both distribution, i.e. we are observing for every theoretical individual in our sample both their potential earnings as an accountant and their potential earnings as an economist.

```
set.seed(10042018)



samples <- 1000000  # Sample size or size of population
cor <- 0.9 # correlation coefficient

# Generate correlated std normal random sample using the mvnorm function from MASS package
data <- mvrnorm(n=samples, mu=c(0, 0), Sigma=matrix(c(1, cor, cor, 1), nrow=2), empirical=FALSE)

# Convert to dataframe
df1 <- as_data_frame(data)

# Make the dataframe easier to use
```

```
df1 <- df1 %>% dplyr::rename(accnt = V1, econ = V2) # rename columns

# Change distributions from std normal to those specified in lecture:
# Accounting ~ N(65000, 5000)
# Economics ~ N(60000, 10000)
mu_econ <- 60000
sigma_econ <- 10000

mu_accnt <- 65000
sigma_accnt <- 5000

df1 <- df1 %>% mutate(
        accnt = accnt*sigma_accnt + mu_accnt, # update accounting variable
        econ = econ*sigma_econ + mu_econ # update econ variable
)
```

Sanity Checks

Let's run a few checks to make sure data looks reasonable. First we'll use the `head` and `tail` commands to get a look at the data. Then, we will make sure that the correlation between the accoutant and economist distributions is 0.86, the value we set when we generated the data, using the `cor` command. Finally, we will run the `summarise` command to see the mean, min, max, and quartiles of each of the distributions.

```
# Look at the first and last 6 rows of the dataframe
head(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 65037. 58736.
## 2 66280. 65719.
## 3 64605. 61839.
## 4 65908. 63792.
## 5 58921. 49206.
## 6 63169. 48734.
```

```
tail(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 58520. 43816.
## 2 68335. 64071.
## 3 62921. 54838.
## 4 61847. 51383.
## 5 70867. 69141.
## 6 68559. 68678.
```

```
# Compare the correlation we set to the correlation we calculate
check <- round(cor(df1$accnt, df1$econ)) == round(cor)
print(ifelse(check, "The correlations are the same!", "Oops, the correlations are not the same."))
```

```
## [1] "The correlations are the same!"
```

```r
# Generate summary statistics
print("Economists Summary Stats")
```

```
## [1] "Economists Summary Stats"
```

```r
summary(df1$econ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11271   53253   59997   59997   66726  106422
```

```r
print("Accountant Summary Stats")
```

```
## [1] "Accountant Summary Stats"
```

```r
summary(df1$accnt)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   40451   61637   64999   65002   68365   90649
```

```r
### Picking a Career

# Our assumption is that every person will choose to be an accountant or an
# economist based on what will maximize their salary. We will assign job labels
# strictly based on where an # individual will earn more,
# i.e. Y<sub>i</sub> = max(Y<sub>e,i</sub>, Y<sub>a,i</sub>)
#```{r jobs}
# Assign job labels
df1 <- df1 %>% mutate(job = ifelse(econ > accnt, "econ", "accnt"))

# Take a look at the change to the dataframe using head
head(df1)
```

```
## # A tibble: 6 x 3
##    accnt   econ job
##    <dbl>  <dbl> <chr>
## 1 65037. 58736. accnt
## 2 66280. 65719. accnt
## 3 64605. 61839. accnt
## 4 65908. 63792. accnt
## 5 58921. 49206. accnt
## 6 63169. 48734. accnt
```

Results

```r
# Make results dataframe
results <- df1 %>%
            group_by(job) %>% # Group all of the rows with the same "job" together
            #summarise('Economist Earnings' = mean(econ), 'Accountant Earnings' = mean(accnt),


              #Changing the code to calculate and display standard deviation
              summarise('Economist Earnings' = sd(econ), 'Accountant Earnings' = sd(accnt),

                      n=n()) %>% # Calculate sd and counts for economicts and accountants
            mutate(job = c("Accountant", "Economist")) %>% # add labels
            t() # transpose

colnames(results) <- c('Accountant', 'Economist')




results <- as.data.frame(results) %>%
            slice(2:4) %>%
            mutate(
              x= c('Economist Earnings', 'Accountant Earnings', 'n')
            ) %>%
            dplyr::select(x, Accountant, Economist)

results
```

```
##                                       x Accountant Economist
## Economist Earnings   Economist Earnings   7997.876  5709.471
## Accountant Earnings Accountant Earnings    4496.64   4020.58
## n                                     n     801309    198691
```

The standard normal sample size is 1000000 It is observed that out of the sample size for economists only 198691 have become economists and hence the standard deviation we observed (5709.471) is significantly less compared to the conditional standard deviation of 100000 The same goes for accountants, out of the sample size for accountants its observed that 801309 have become accountants, hence the standard deviation we observed (4496.64) is less compared to the conditional standard deviation set of 5000

Roy Model Simulation 4

Generate Random Data

Example: Economist earnings are normally distributed with mean = 60k and stdev = 10k, and Accountant earnings are normally distributed with mean = 65k and st dev = 5k. This generation will produce data that pairs draws from both distribution, i.e. we are observing for every theoretical individual in our sample both their potential earnings as an accountant and their potential earnings as an economist.

```r
set.seed(10042018)


samples <- 1000000  # Sample size or size of population
cor <- 0.9 # correlation coefficient
```

```
# Generate correlated std normal random sample using the mvnorm function from MASS package
data <- mvrnorm(n=samples, mu=c(0, 0), Sigma=matrix(c(1, cor, cor, 1), nrow=2), empirical=FALSE)

# Convert to dataframe
df1 <- as_data_frame(data)

# Make the dataframe easier to use
df1 <- df1 %>% dplyr::rename(accnt = V1, econ = V2) # rename columns

# Change distributions from std normal to those specified in lecture:
# Accounting ~ N(65000, 5000)
# Economics ~ N(60000, 10000)
mu_econ <- 60000

  sigma_econ <- 12000

mu_accnt <- 65000
sigma_accnt <- 5000

df1 <- df1 %>% mutate(
        accnt = accnt*sigma_accnt + mu_accnt, # update accounting variable
        econ = econ*sigma_econ + mu_econ # update econ variable
)
```

Sanity Checks

Let's run a few checks to make sure data looks reasonable. First we'll use the `head` and `tail` commands to get a look at the data. Then, we will make sure that the correlation between the accoutant and economist distributions is 0.86, the value we set when we generated the data, using the `cor` command. Finally, we will run the `summarise` command to see the mean, min, max, and quartiles of each of the distributions.

```
# Look at the first and last 6 rows of the dataframe
head(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 65037. 58484.
## 2 66280. 66863.
## 3 64605. 62207.
## 4 65908. 64550.
## 5 58921. 47047.
## 6 63169. 46481.
```

```
tail(df1)
```

```
## # A tibble: 6 x 2
##    accnt   econ
##    <dbl>  <dbl>
## 1 58520. 40580.
## 2 68335. 64885.
## 3 62921. 53806.
## 4 61847. 49660.
## 5 70867. 70969.
## 6 68559. 70414.
```

```r
# Compare the correlation we set to the correlation we calculate
check <- round(cor(df1$accnt, df1$econ)) == round(cor)
print(ifelse(check, "The correlations are the same!", "Oops, the correlations are not the same."))
```

```
## [1] "The correlations are the same!"
```

```r
# Generate summary statistics
print("Economists Summary Stats")
```

```
## [1] "Economists Summary Stats"
```

```r
summary(df1$econ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1525   51903   59996   59997   68071  115707
```

```r
print("Accountant Summary Stats")
```

```
## [1] "Accountant Summary Stats"
```

```r
summary(df1$accnt)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   40451   61637   64999   65002   68365   90649
```

```r
### Picking a Career

#Our assumption is that every person will choose to be an accountant or an
#economist based on what will maximize their salary. We will assign job labels
#strictly based on where an individual will earn more,
#i.e. Y<sub>i</sub> = max(Y<sub>e,i</sub>, Y<sub>a,i</sub>)
#```{r jobs}
# Assign job labels
df1 <- df1 %>% mutate(job = ifelse(econ > accnt, "econ", "accnt"))

# Take a look at the change to the dataframe using head
head(df1)
```

```
## # A tibble: 6 x 3
##    accnt   econ job
##    <dbl>  <dbl> <chr>
## 1 65037. 58484. accnt
## 2 66280. 66863. econ
## 3 64605. 62207. accnt
## 4 65908. 64550. accnt
## 5 58921. 47047. accnt
## 6 63169. 46481. accnt
```

Results

```
# Make results dataframe
results <- df1 %>%
            group_by(job) %>% # Group all of the rows with the same "job" together
            summarise('Economist Earnings' = mean(econ), 'Accountant Earnings' = mean(accnt),


                      n=n()) %>% # Calculate means and counts for economicts and accountants
            mutate(job = c("Accountant", "Economist")) %>% # add labels
            t() # transpose

colnames(results) <- c('Accountant', 'Economist')




results <- as.data.frame(results) %>%
            slice(2:4) %>%
            mutate(
              x= c('Economist Earnings', 'Accountant Earnings', 'n')
            ) %>%
            dplyr::select(x, Accountant, Economist)

results
```

```
##                                        x Accountant Economist
## Economist Earnings   Economist Earnings   54934.66  74370.56
## Accountant Earnings Accountant Earnings   63371.95  69632.12
## n                                      n     739539    260461
```

The data from this simulation shows an increase in total economists and decrease in accountants. As the standard deviation of economist earnings is increased from $10000 to $12000, the economist earnings of economist have increased and accountants with economist earnings have reduced. This is due to the flattening of the curve that happens because of the increase in standard deviation causing the occurrences of high economist earnings has resulted accountants becoming economists, which means increase in total economists and decrease in total accountants. Due to the same, the mean of economists with economist earnings increases (Mean changed from 73066.65 to 74370.56) whereas the mean of accountants with accountant earn ings has reduced(Mean changed from 63822.98 to 63371.95). One another observation is the increase in people with economists earnings (198340 to 260672) and reduction in number of people with accountants earnings (801660 to 739328)